

Package ‘okxr’

August 1, 2026

Title R Interface to the 'OKX' REST API

Version 0.4.8

Author Oliver Zhou [aut, cre],
Lily Li [aut]

Maintainer Oliver Zhou <oliver.yxzhou@gmail.com>

Description Provides lightweight R wrappers for the 'OKX' REST API, covering endpoints for market data, trading, account management, asset balances, and copy trading. The upstream API reference is available at <<https://www.okx.com/docs-v5/en/>>.

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Imports rlang, httr, jsonlite, digest, base64enc, data.table

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Collate okxr-package.R def_constants.R utils_request_helpers.R
utils_auth.R utils_parser.R utils_get_generator.R
utils_post_generator.R utils_labels.R wrappers_get_market.R
wrappers_get_asset.R wrappers_get_account.R
wrappers_get_trade.R wrappers_get_copy_trade.R
wrappers_post_asset.R wrappers_post_configure.R
wrappers_post_trade.R

URL <https://github.com/OliverLDS/okxr>

BugReports <https://github.com/OliverLDS/okxr/issues>

RoxygenNote 8.0.0

NeedsCompilation no

Repository CRAN

Date/Publication 2026-08-01 13:50:02 UTC

Contents

okxr-package	5
.build_request	6
.execute_get_action	7
.execute_post_action	7
.get_headers	8
get_account_adjust_leverage_info	8
get_account_balance	9
get_account_bills	10
get_account_bills_archive	11
get_account_bills_history_archive	12
get_account_bill_types	12
get_account_collateral_assets	13
get_account_config	14
get_account_greeks	15
get_account_instruments	15
get_account_interest_accrued	16
get_account_interest_limits	17
get_account_interest_rate	18
get_account_leverage_info	18
get_account_max_avail_size	19
get_account_max_loan	20
get_account_max_size	21
get_account_max_withdrawal	22
get_account_mmp_config	22
get_account_move_positions_history	23
get_account_positions	24
get_account_positions_history	25
get_account_position_risk	26
get_account_position_tiers	26
get_account_precheck_set_delta_neutral	27
get_account_risk_state	28
get_account_set_account_switch_precheck	28
get_account_spot_borrow_repay_history	29
get_account_subaccount_balances	29
get_account_subaccount_max_withdrawal	30
get_account_subtypes	31
get_account_trade_fee	31
get_asset_asset_valuation	32
get_asset_balances	33
get_asset_bills	33
get_asset_bills_history	34
get_asset_convert_currencies	35
get_asset_convert_currency_pair	36
get_asset_convert_history	36
get_asset_currencies	37
get_asset_deposit_address	38

get_asset_deposit_history	38
get_asset_deposit_withdraw_status	39
get_asset_exchange_list	40
get_asset_non_tradable_assets	40
get_asset_transfer_state	41
get_asset_withdrawal_history	41
get_copy_trade_config	42
get_copy_trade_current_subpos	43
get_copy_trade_historical_subpos	44
get_copy_trade_instruments	45
get_copy_trade_my_leaders	46
get_copy_trade_profit_sharing_details	47
get_copy_trade_public_config	48
get_copy_trade_public_copy_traders	48
get_copy_trade_public_current_subpositions	49
get_copy_trade_public_lead_traders	50
get_copy_trade_public_pnl	51
get_copy_trade_public_preference_currency	51
get_copy_trade_public_stats	52
get_copy_trade_public_subpositions_history	53
get_copy_trade_public_weekly_pnl	53
get_copy_trade_settings	54
get_copy_trade_total_profit_sharing	55
get_copy_trade_total_unrealized_profit_sharing	56
get_copy_trade_unrealized_profit_sharing_details	56
get_market_block_ticker	57
get_market_block_tickers	57
get_market_books	58
get_market_books_rpi	59
get_market_candles	59
get_market_exchange_rate	60
get_market_history_candles	61
get_market_history_index_candles	62
get_market_history_mark_price_candles	63
get_market_history_trades	64
get_market_index_candles	65
get_market_index_components	66
get_market_index_tickers	66
get_market_mark_price_candles	67
get_market_option_instrument_family_trades	68
get_market_platform_24_volume	68
get_market_ticker	69
get_market_tickers	69
get_market_trades	70
get_public_block_trades	71
get_public_convert_contract_coin	71
get_public_delivery_exercise_history	72
get_public_discount_rate_interest_free_quota	73

get_public_economic_calendar	74
get_public_estimated_price	74
get_public_estimated_settlement_info	75
get_public_event_contract_events	76
get_public_event_contract_markets	77
get_public_event_contract_series	78
get_public_funding_rate	78
get_public_funding_rate_history	79
get_public_instruments	80
get_public_instrument_tick_bands	81
get_public_insurance_fund	82
get_public_interest_rate_loan_quota	83
get_public_mark_price	83
get_public_mm_instrument_types	84
get_public_open_interest	85
get_public_option_trades	86
get_public_opt_summary	87
get_public_position_tiers	87
get_public_premium_history	88
get_public_price_limit	89
get_public_settlement_history	90
get_public_time	90
get_public_underlying	91
get_trade_account_rate_limit	91
get_trade_easy_convert_currency_list	92
get_trade_easy_convert_history	93
get_trade_fills	93
get_trade_fills_history	94
get_trade_one_click_repay_currency_list	95
get_trade_one_click_repay_currency_list_v2	96
get_trade_one_click_repay_history	97
get_trade_one_click_repay_history_v2	97
get_trade_order	98
get_trade_orders_algo_history	99
get_trade_orders_algo_pending	100
get_trade_orders_history	101
get_trade_orders_history_7d	102
get_trade_orders_history_archive	103
get_trade_orders_pending	104
get_trade_order_algo	105
get_user_glp_historical_performance	106
get_user_glp_today_performance	107
get_var_label	107
post_account_account_level_switch_preset	108
post_account_mmp_config	109
post_account_mmp_reset	109
post_account_move_positions	110
post_account_position_margin_balance	111

post_account_set_account_level	112
post_account_set_auto_loan	112
post_account_set_auto_repay	113
post_account_set_collateral_assets	113
post_account_set_fee_type	114
post_account_set_greeks	115
post_account_set_leverage	115
post_account_set_position_mode	116
post_account_spot_manual_borrow_repay	117
post_asset_cancel_withdrawal	117
post_asset_convert_estimate_quote	118
post_asset_convert_trade	119
post_asset_transfer	120
post_asset_withdrawal	121
post_trade_amend_algos	122
post_trade_amend_batch_orders	123
post_trade_amend_order	124
post_trade_batch_orders	125
post_trade_cancel_algos	125
post_trade_cancel_all_after	126
post_trade_cancel_batch_orders	127
post_trade_cancel_order	127
post_trade_close_position	128
post_trade_mass_cancel	129
post_trade_order	129
post_trade_order_algo	131
post_trade_order_precheck	132
set_okxr_options	134

Index**135**

okxr-package

*okxr: R Interface to the OKX REST API***Description**

‘okxr’ provides lightweight wrappers for selected OKX REST API endpoints, including market data, account information, asset metadata, order-book trade queries, trading actions, and copy-trading endpoints.

Details

Public market and public reference endpoints can be called without credentials. Private account, asset, trade, and copy-trading endpoints require an OKX API credential list with ‘api_key’, ‘secret_key’, and ‘passphrase’ entries.

If ‘config\$demo’ is ‘TRUE’, signed requests include OKX’s simulated trading header. Request timeout defaults to 10 seconds and can be set globally with ‘set_okxr_options(timeout = 15)’ or per request with ‘config\$timeout’.

By default, wrappers return parsed 'data.table' objects. Use 'set_okxr_options(raw_data = TRUE)' to return raw API 'data' payloads instead.

Network failures, request timeouts, OKX error responses, or empty API 'data' payloads may return 'NULL' with a warning. Live API examples are intentionally non-running because they require credentials, network access, and may have account-specific side effects.

Author(s)

Maintainer: Oliver Zhou <oliver.yxzhou@gmail.com>

Authors:

- Oliver Zhou <oliver.yxzhou@gmail.com>
- Lily Li <lilyli.qc.hk@gmail.com>

See Also

[set_okxr_options()], [get_market_candles()], [post_trade_order()]

.build_request	<i>Build a full OKX request object</i>
----------------	--

Description

Assemble URL, headers, and body for an OKX API call.

Usage

```
.build_request(
  htr_method,
  base_url,
  api_path,
  query_string,
  config = NULL,
  body_json = "",
  auth = TRUE
)
```

Arguments

htr_method	HTTP method ("GET" or "POST").
base_url	Base URL of the OKX API.
api_path	API path (e.g., "/api/v5/account/balance").
query_string	Query string starting with "?" or empty.
config	List with API credentials.
body_json	Optional JSON string for POST body.
auth	Logical. Whether to sign the request with OKX credentials.

Value

A list with elements: 'method', 'url', 'full_path', 'headers', and 'body_json'.

.execute_get_action	Execute a GET request to OKX
---------------------	------------------------------

Description

Perform a signed GET call to the OKX API.

Usage

```
.execute_get_action(api_path, query_string, config = NULL, auth = TRUE)
```

Arguments

api_path	API path (e.g., "/api/v5/account/balance").
query_string	Query string starting with "?" or empty.
config	List with API credentials.
auth	Logical. Whether to sign the request with OKX credentials.

Value

An 'httr' response object, or 'NULL' if the request fails.

.execute_post_action	Execute a POST request to OKX
----------------------	-------------------------------

Description

Perform a signed POST call to the OKX API.

Usage

```
.execute_post_action(api_path, body_list, config)
```

Arguments

api_path	API path (e.g., "/api/v5/trade/order").
body_list	List to be converted to JSON for the request body.
config	List with API credentials.

Value

An 'httr' response object, or 'NULL' if the request fails.

<code>.get_headers</code>	<i>Create OKX API request headers</i>
---------------------------	---------------------------------------

Description

Generate the required signed headers for an OKX REST request.

Usage

```
.get_headers(config, http_method, http_path, body_json = "")
```

Arguments

<code>config</code>	List with 'api_key', 'secret_key', and 'passphrase'.
<code>http_method</code>	HTTP method as a string (e.g., "GET" or "POST").
<code>http_path</code>	Path portion of the API endpoint.
<code>body_json</code>	Optional JSON string of the request body.

Value

A 'http::add_headers' object containing the signed headers.

<code>get_account_adjust_leverage_info</code>	<i>Get account leverage adjustment estimate</i>
---	---

Description

Estimate account effects under a target leverage.

Usage

```
get_account_adjust_leverage_info(
  inst_type,
  mgn_mode,
  lever,
  inst_id = NULL,
  ccy = NULL,
  pos_side = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character. Instrument type: "MARGIN", "SWAP", or "FUTURES".
mgn_mode	Character. Margin mode: "cross" or "isolated".
lever	Character or numeric. Target leverage.
inst_id	Character or 'NULL'. Instrument ID.
ccy	Character or 'NULL'. Margin currency.
pos_side	Character or 'NULL'. Position side.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A one-row 'data.frame' with estimated leverage-adjustment metrics.

get_account_balance	<i>Get account balance</i>
---------------------	----------------------------

Description

Retrieve account-level margin and equity information for your OKX account.

Usage

```
get_account_balance(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'. May also include 'base_url'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

This wraps '/api/v5/account/balance'. Returns one row per account-level equity snapshot. Timestamps are parsed into 'POSIXct' in the given 'tz'.

Value

A 'data.frame' with account balance and margin metrics (e.g., 'totalEq', 'isoEq', 'adjEq', 'availEq', 'ordFroz', 'imr', 'mmr', 'upl', 'mgnRatio', ...). Timestamp columns ('uTime') are 'POSIXct'.

Note

Since okxr 0.1.1

See Also

[get_account_positions()], [get_account_leverage_info()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
bal <- get_account_balance(config = cfg)
head(bal)

## End(Not run)
```

get_account_bills	<i>Get account bills</i>
-------------------	--------------------------

Description

Retrieve account bill details from the last 7 days.

Usage

```
get_account_bills(
  inst_type = NULL,
  ccy = NULL,
  mgn_mode = NULL,
  ct_type = NULL,
  type = NULL,
  sub_type = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
ccy	Character or 'NULL'. Currency filter.
mgn_mode	Character or 'NULL'. Margin mode filter.
ct_type	Character or 'NULL'. Contract type filter.
type	Character or 'NULL'. Bill type filter.
sub_type	Character or 'NULL'. Bill subtype filter.
after	Character or 'NULL'. Pagination cursor for earlier records.

before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with account bill rows.

```
get_account_bills_archive
```

Get archived account bills

Description

Retrieve archived account bill details.

Usage

```
get_account_bills_archive(
  inst_type = NULL,
  ccy = NULL,
  mgn_mode = NULL,
  ct_type = NULL,
  type = NULL,
  sub_type = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
ccy	Character or 'NULL'. Currency filter.
mgn_mode	Character or 'NULL'. Margin mode filter.
ct_type	Character or 'NULL'. Contract type filter.
type	Character or 'NULL'. Bill type filter.
sub_type	Character or 'NULL'. Bill subtype filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with archived account bill rows.

```
get_account_bills_history_archive
```

Get archived account bill export links

Description

Retrieve the generated CSV export link for historical account bills since 2021.

Usage

```
get_account_bills_history_archive(
  year,
  quarter,
  type = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

year	Character or numeric. Four-digit year.
quarter	Character. Quarter code, one of "Q1" to "Q4".
type	Character or 'NULL'. Optional comma-separated bill type filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with file-link status rows.

```
get_account_bill_types
```

Get account bill types

Description

Alias for [get_account_subtypes()] using OKX's current endpoint label.

Usage

```
get_account_bill_types(type = NULL, config, tz = .okx_default_tz)
```

Arguments

type	Character or 'NULL'. Bill type filter. Multiple values may be provided as a comma-separated string.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with bill type descriptions and JSON-encoded 'subTypeDetails'.

```
get_account_collateral_assets
```

Get account collateral assets

Description

Retrieve collateral-enabled status for one or more currencies.

Usage

```
get_account_collateral_assets(
  ccy = NULL,
  collateral_enabled = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

ccy	Character or 'NULL'. One currency or a comma-separated list of up to 20 currencies.
collateral_enabled	Logical, character, or 'NULL'. Filter by collateral status.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with currency-level collateral flags.

get_account_config	<i>Get account configuration</i>
--------------------	----------------------------------

Description

Retrieve account-level configuration information.

Usage

```
get_account_config(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/account/config'. Includes account ID, mode, and position mode flags. Returns one row.

Value

A 'data.frame' with columns like 'uid', 'mainUid', 'acctLv', 'posMode', 'autoLoan', etc.

Note

Since okxr 0.1.2

See Also

[get_account_balance()], [get_account_leverage_info()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
cfg_info <- get_account_config(config = cfg)
cfg_info

## End(Not run)
```

get_account_greeks	<i>Get account Greeks</i>
--------------------	---------------------------

Description

Retrieve currency-level Greeks across the account.

Usage

```
get_account_greeks(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with one row per currency and Greek metrics in both Black-Scholes and coin terms.

get_account_instruments	<i>Get account-available instruments</i>
-------------------------	--

Description

Retrieve available instruments for the current account and instrument type.

Usage

```
get_account_instruments(  
  inst_type,  
  uly = NULL,  
  inst_family = NULL,  
  inst_id = NULL,  
  series_id = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type. One of "SPOT", "MARGIN", "SWAP", "FUTURES", "OPTION", or "EVENTS".
uly	Character or 'NULL'. Underlying, where applicable.
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Specific instrument ID filter.
series_id	Character or 'NULL'. Event-contract series ID. Required by OKX when 'inst_type' = "EVENTS".
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/account/instruments'. This endpoint is account-scoped and may return a subset of instruments available to the authenticated account.

Value

A 'data.frame' with account-available instrument metadata, including identifiers, currencies, tick size, lot size, leverage, expiry/listing timestamps, and state where returned by OKX.

```
get_account_interest_accrued
```

Get account interest accrued history

Description

Retrieve accrued borrowing interest records for the past year.

Usage

```
get_account_interest_accrued(
  type = NULL,
  ccy = NULL,
  inst_id = NULL,
  mgn_mode = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

type	Character or 'NULL'. Loan type. Currently "2" for market loans.
ccy	Character or 'NULL'. Loan currency.
inst_id	Character or 'NULL'. Instrument ID.
mgn_mode	Character or 'NULL'. Margin mode.
after	Character or 'NULL'. Pagination cursor for earlier rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per accrued-interest event.

```
get_account_interest_limits
```

Get account borrow interest and limits

Description

Retrieve account-level debt, next accrual timestamps, and nested per-currency borrowing-limit records.

Usage

```
get_account_interest_limits(
  type = NULL,
  ccy = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

type	Character or 'NULL'. Loan type. Currently "2" for market loans.
ccy	Character or 'NULL'. Loan currency.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with account-level limit fields; nested per-currency 'records' are JSON-encoded.

get_account_interest_rate

Get account borrowing interest rates

Description

Retrieve the current leveraged currency borrowing market interest rate for one currency or for all eligible currencies.

Usage

```
get_account_interest_rate(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. Currency filter, e.g. "BTC".
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'ccy' and 'interestRate'.

get_account_leverage_info

Get account leverage settings

Description

Retrieve leverage configuration for a given instrument and margin mode.

Usage

```
get_account_leverage_info(inst_id, mgn_mode, config, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT".
mgn_mode	Character. Margin mode. One of "cross" or "isolated".
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps `/api/v5/account/leverage-info`. Requires both `'inst_id'` and `'mgn_mode'`. Returns current leverage values (numeric).

Value

A `'data.frame'` with columns `'instId'`, `'mgnMode'`, `'posSide'`, and `'lever'`.

Note

Since okxr 0.1.1

See Also

`[get_account_balance()]`, `[get_account_positions()]`

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_account_leverage_info(
  inst_id = "BTC-USDT",
  mgn_mode = "cross",
  config = cfg
)

## End(Not run)
```

`get_account_max_avail_size`

Get maximum available tradable amount

Description

Retrieve the maximum available buy and sell amount for an instrument under the requested trade mode.

Usage

```
get_account_max_avail_size(
  inst_id,
  td_mode,
  ccy = NULL,
  reduce_only = NULL,
  px = NULL,
  trade_quote_ccy = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_id	Character. One instrument ID or a comma-separated list of up to five IDs.
td_mode	Character. Trade mode: "cross", "isolated", "cash", or "spot_isolated".
ccy	Character or 'NULL'. Margin currency where applicable.
reduce_only	Logical, character, or 'NULL'. Whether to reduce position only. Only applicable to margin endpoints that support it.
px	Character, numeric, or 'NULL'. Optional closing price, when supported by OKX.
trade_quote_ccy	Character or 'NULL'. Quote currency used for trading for spot instruments.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'availBuy', and 'availSell'.

get_account_max_loan	<i>Get account maximum loan</i>
----------------------	---------------------------------

Description

Retrieve the maximum loan for manual borrow or margin borrowing scenarios.

Usage

```
get_account_max_loan(
  mgn_mode,
  inst_id = NULL,
  ccy = NULL,
  mgn_ccy = NULL,
  trade_quote_ccy = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

mgn_mode	Character. Margin mode: "cross" or "isolated".
inst_id	Character or 'NULL'. Instrument ID(s).
ccy	Character or 'NULL'. Currency.
mgn_ccy	Character or 'NULL'. Margin currency.
trade_quote_ccy	Character or 'NULL'. Quote currency for trading.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'mgnMode', 'mgnCcy', 'maxLoan', 'ccy', and 'side'.

get_account_max_size	<i>Get maximum order size</i>
----------------------	-------------------------------

Description

Retrieve the maximum order quantity allowed for one or more instruments under the requested trade mode.

Usage

```
get_account_max_size(
  inst_id,
  td_mode,
  ccy = NULL,
  px = NULL,
  leverage = NULL,
  trade_quote_ccy = NULL,
  outcome = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_id	Character. One instrument ID or a comma-separated list of up to five IDs in the same instrument type.
td_mode	Character. Trade mode: "cross", "isolated", "cash", or "spot_isolated".
ccy	Character or 'NULL'. Margin currency where applicable.
px	Character, numeric, or 'NULL'. Optional price override.
leverage	Character, numeric, or 'NULL'. Optional leverage override.
trade_quote_ccy	Character or 'NULL'. Quote currency used for trading for spot instruments.
outcome	Character or 'NULL'. Events market outcome, "yes" or "no".
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'ccy', 'maxBuy', and 'maxSell'.

```
get_account_max_withdrawal
```

Get account maximum withdrawals

Description

Retrieve the maximum transferable amount from trading to funding account.

Usage

```
get_account_max_withdrawal(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. One currency or a comma-separated list of up to 20 currencies.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with per-currency maximum withdrawal values.

```
get_account_mmp_config
```

Get account MMP configuration

Description

Retrieve option-market-maker-protection configuration for one or more instrument families.

Usage

```
get_account_mmp_config(inst_family = NULL, config, tz = .okx_default_tz)
```

Arguments

inst_family	Character or 'NULL'. Instrument family filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with MMP configuration fields.

`get_account_move_positions_history`*Get account move positions history*

Description

Retrieve move-position requests from the last three days.

Usage

```
get_account_move_positions_history(  
    block_td_id = NULL,  
    client_id = NULL,  
    begin_ts = NULL,  
    end_ts = NULL,  
    limit = NULL,  
    state = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

<code>block_td_id</code>	Character or 'NULL'. OKX block trade identifier.
<code>client_id</code>	Character or 'NULL'. Client-supplied identifier.
<code>begin_ts</code>	Character or 'NULL'. Inclusive start timestamp in milliseconds.
<code>end_ts</code>	Character or 'NULL'. Inclusive end timestamp in milliseconds.
<code>limit</code>	Integer or 'NULL'. Number of rows to request.
<code>state</code>	Character or 'NULL'. Transfer state filter, "filled" or "pending".
<code>config</code>	List. API credentials/config.
<code>tz</code>	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with top-level move-position metadata; nested 'legs' are JSON-encoded.

get_account_positions *Get account open positions*

Description

Retrieve all currently open positions under the account.

Usage

```
get_account_positions(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/account/positions'. Returns one row per open position.

Value

A 'data.frame' with columns such as 'instId', 'posId', 'posSide', 'pos', 'lever', 'avgPx', 'markPx', 'upl', 'realizedPnl', etc. Timestamps ('cTime', 'uTime') are 'POSIXct'.

Note

Since okxr 0.1.1

See Also

[get_account_balance()], [get_account_positions_history()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
pos <- get_account_positions(config = cfg)
pos

## End(Not run)
```

get_account_positions_history
Get account position history

Description

Retrieve historical records of closed or adjusted positions.

Usage

```
get_account_positions_history(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/account/positions-history'. Includes closed positions and their realized PnL. Returns one row per historical record.

Value

A 'data.frame' with columns such as 'instId', 'posId', 'posSide', 'pos', 'lever', 'realizedPnl', 'fee', plus timestamp fields ('cTime', 'uTime').

Note

Since okxr 0.1.1

See Also

[get_account_positions()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
hist <- get_account_positions_history(config = cfg)
tail(hist)

## End(Not run)
```

```
get_account_position_risk
```

Get account and position risk snapshot

Description

Retrieve account-level adjusted equity together with same-snapshot balance and position risk payloads.

Usage

```
get_account_position_risk(inst_type = NULL, config, tz = .okx_default_tz)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter. One of "MARGIN", "SWAP", "FUTURES", or "OPTION".
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per OKX risk snapshot. Nested balance and position payloads are returned as JSON strings in 'balData' and 'posData'.

```
get_account_position_tiers
```

Get account position tiers

Description

Retrieve portfolio-margin position limits for one or more instrument families.

Usage

```
get_account_position_tiers(
  inst_type,
  inst_family,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character. Instrument type: "SWAP", "FUTURES", or "OPTION".
inst_family	Character. One instrument family or a comma-separated list of up to five families.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with instrument-family position limits.

get_account_precheck_set_delta_neutral
<i>Precheck delta-neutral strategy switch</i>

Description

Retrieve unmatched information that blocks switching into the requested strategy type.

Usage

```
get_account_precheck_set_delta_neutral(stgy_type, config, tz = .okx_default_tz)
```

Arguments

stgy_type	Character or numeric. Strategy type. "0" for general or "1" for delta neutral.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A one-row 'data.frame' with JSON-encoded unmatched information.

`get_account_risk_state`*Get account risk state*

Description

Retrieve portfolio-margin account risk flags and affected risk units.

Usage

```
get_account_risk_state(config, tz = .okx_default_tz)
```

Arguments

<code>config</code>	List. API credentials/config.
<code>tz</code>	Character. Time zone for parsing timestamps. Default <code>"Asia/Hong_Kong"</code> .

Value

A one-row `'data.frame'` with the account risk flag and JSON-encoded risk-unit arrays.

`get_account_set_account_switch_precheck`*Precheck account mode switch*

Description

Retrieve precheck information and any unmatched requirements for switching to a target account mode.

Usage

```
get_account_set_account_switch_precheck(acct_lv, config, tz = .okx_default_tz)
```

Arguments

<code>acct_lv</code>	Character or numeric. Target account mode.
<code>config</code>	List. API credentials/config.
<code>tz</code>	Character. Time zone for parsing timestamps. Default <code>"Asia/Hong_Kong"</code> .

Value

A `'data.frame'` with switch-precheck fields; nested margin and unmatched-information structures are JSON-encoded.

```
get_account_spot_borrow_repay_history
```

Get spot borrow and repay history

Description

Retrieve spot-mode borrow and repay history.

Usage

```
get_account_spot_borrow_repay_history(  
    ccy = NULL,  
    type = NULL,  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
type	Character or 'NULL'. Event type filter.
after	Character or 'NULL'. Pagination cursor for earlier rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per borrow or repay event.

```
get_account_subaccount_balances
```

Get sub-account trading balances

Description

Retrieve account-level trading balances for a sub-account from the master account.

Usage

```
get_account_subaccount_balances(sub_acct, config, tz = .okx_default_tz)
```

Arguments

sub_acct	Character. Sub-account name.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default <code>"Asia/Hong_Kong"</code> .

Value

A `'data.frame'` with account-level balance fields; nested per-currency `'details'` are JSON-encoded.

```
get_account_subaccount_max_withdrawal
```

Get sub-account maximum withdrawals

Description

Retrieve the maximum withdrawal information for a sub-account from the master account.

Usage

```
get_account_subaccount_max_withdrawal(
  sub_acct,
  ccy = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

sub_acct	Character. Sub-account name.
ccy	Character or <code>'NULL'</code> . One currency or a comma-separated list of up to 20 currencies.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default <code>"Asia/Hong_Kong"</code> .

Value

A `'data.frame'` with per-currency maximum withdrawal values.

get_account_subtypes *Get account bill subtypes*

Description

Retrieve available account bill types and subtype descriptions.

Usage

```
get_account_subtypes(type = NULL, config, tz = .okx_default_tz)
```

Arguments

type	Character or 'NULL'. Bill type filter. Multiple values may be provided as a comma-separated string.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with bill type descriptions and JSON-encoded 'subTypeDetails'.

get_account_trade_fee *Get account trade fee rates*

Description

Retrieve the account's trade fee schedule for a specific instrument type and optional instrument, instrument family, or trading fee group.

Usage

```
get_account_trade_fee(  
  inst_type,  
  inst_id = NULL,  
  inst_family = NULL,  
  group_id = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type. One of "SPOT", "MARGIN", "SWAP", "FUTURES", "OPTION", or "EVENTS".
inst_id	Character or 'NULL'. Instrument ID for spot or margin.
inst_family	Character or 'NULL'. Instrument family for futures, swaps, or options.
group_id	Character or 'NULL'. Trading fee group ID. Do not combine with 'inst_id' or 'inst_family'.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with fee level, fee-rate columns, and any nested 'feeGroup' or deprecated 'fiat' details JSON-encoded as strings.

```
get_asset_asset_valuation
```

Get asset valuation

Description

Retrieve total account valuation for funding assets.

Usage

```
get_asset_asset_valuation(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. Valuation currency filter.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with valuation summary rows.

get_asset_balances	<i>Get asset balances</i>
--------------------	---------------------------

Description

Retrieves the available, total, and frozen balance for each asset in the account.

Usage

```
get_asset_balances(config, tz = .okx_default_tz)
```

Arguments

config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string (default: "Asia/Hong_Kong").

Value

A data.frame with balances per currency.

get_asset_bills	<i>Get asset bills</i>
-----------------	------------------------

Description

Retrieve recent funding account bills.

Usage

```
get_asset_bills(  
  ccy = NULL,  
  type = NULL,  
  client_id = NULL,  
  third_party_type = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
type	Character or 'NULL'. Bill type filter.
client_id	Character or 'NULL'. Client-supplied transfer or withdrawal ID.
third_party_type	Character or 'NULL'. Third-party bill type filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with asset bill rows.

get_asset_bills_history	<i>Get asset bills history</i>
-------------------------	--------------------------------

Description

Retrieve historical funding account bills.

Usage

```
get_asset_bills_history(
  ccy = NULL,
  type = NULL,
  client_id = NULL,
  third_party_type = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  paging_type = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
type	Character or 'NULL'. Bill type filter.
client_id	Character or 'NULL'. Client-supplied transfer or withdrawal ID.
third_party_type	Character or 'NULL'. Third-party bill type filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
paging_type	Character or 'NULL'. Paging type selector.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with historical asset bill rows.

get_asset_convert_currencies
Get convert currencies

Description

Retrieve currencies supported by the asset convert API.

Usage

```
get_asset_convert_currencies(config, tz = .okx_default_tz)
```

Arguments

config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with supported convert currencies.

get_asset_convert_currency_pair
Get convert currency pair

Description

Retrieve convert metadata for a currency pair.

Usage

```
get_asset_convert_currency_pair(  
    from_ccy,  
    to_ccy,  
    convert_mode = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

from_ccy	Character. Currency to convert from.
to_ccy	Character. Currency to convert to.
convert_mode	Character or 'NULL'. Convert mode selector.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with currency-pair convert metadata.

get_asset_convert_history
Get convert history

Description

Retrieve historical asset convert trades.

Usage

```
get_asset_convert_history(
    cl_t_req_id = NULL,
    after = NULL,
    before = NULL,
    limit = NULL,
    tag = NULL,
    config,
    tz = .okx_default_tz
)
```

Arguments

cl_t_req_id	Character or 'NULL'. Client trade request ID.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
tag	Character or 'NULL'. Broker tag filter.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with convert history rows.

get_asset_currencies *Get funding currencies*

Description

Retrieve currencies available to the current account.

Usage

```
get_asset_currencies(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. Single currency or comma-separated currencies.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string (default: "Asia/Hong_Kong").

Value

A data.frame with currency and chain metadata.

```
get_asset_deposit_address
```

Get deposit address

Description

Retrieve deposit addresses for a currency.

Usage

```
get_asset_deposit_address(ccy, config, tz = .okx_default_tz)
```

Arguments

ccy	Character. Currency, e.g. "BTC" or "USDT".
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string (default: "Asia/Hong_Kong").

Value

A data.frame with deposit address rows.

```
get_asset_deposit_history
```

Get asset deposit history

Description

Retrieves a record of all asset deposits made to your account.

Usage

```
get_asset_deposit_history(  
  ccy = NULL,  
  dep_id = NULL,  
  from_wd_id = NULL,  
  tx_id = NULL,  
  type = NULL,  
  state = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
dep_id	Character or 'NULL'. Deposit ID filter.
from_wd_id	Character or 'NULL'. Source withdrawal ID filter.
tx_id	Character or 'NULL'. Transaction hash filter.
type	Character or 'NULL'. Deposit type filter.
state	Character or 'NULL'. Deposit state filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string (default: "Asia/Hong_Kong").

Value

A data.frame with deposit history rows and detailed transfer metadata.

```
get_asset_deposit_withdraw_status
```

Get deposit or withdrawal status

Description

Retrieve detailed status for a deposit or withdrawal.

Usage

```
get_asset_deposit_withdraw_status(
  wd_id = NULL,
  tx_id = NULL,
  ccy = NULL,
  to = NULL,
  chain = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

wd_id	Character or 'NULL'. Withdrawal ID.
tx_id	Character or 'NULL'. Deposit transaction hash.
ccy	Character or 'NULL'. Currency filter used with tx_id.
to	Character or 'NULL'. Destination address used with tx_id.

chain	Character or 'NULL'. Chain identifier used with tx_id.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with detailed deposit/withdraw status rows.

```
get_asset_exchange_list
```

Get exchange list

Description

Retrieve the public exchange list used by asset withdrawal metadata.

Usage

```
get_asset_exchange_list(tz = .okx_default_tz)
```

Arguments

tz	Timezone string.
----	------------------

Value

A data.frame with exchange identifiers and names.

```
get_asset_non_tradable_assets
```

Get non-tradable assets

Description

Retrieve balances and withdrawal metadata for non-tradable assets.

Usage

```
get_asset_non_tradable_assets(ccy = NULL, config, tz = .okx_default_tz)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with non-tradable asset rows.

```
get_asset_transfer_state
```

Get asset transfer state

Description

Retrieve the state of a funding transfer.

Usage

```
get_asset_transfer_state(  
  trans_id = NULL,  
  client_id = NULL,  
  type = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

trans_id	Character or 'NULL'. Transfer ID.
client_id	Character or 'NULL'. Client-supplied transfer ID.
type	Character or 'NULL'. Transfer type.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string.

Value

A data.frame with transfer state rows.

```
get_asset_withdrawal_history
```

Get asset withdrawal history

Description

Retrieves a record of all asset withdrawals from your account.

Usage

```
get_asset_withdrawal_history(
    ccy = NULL,
    wd_id = NULL,
    client_id = NULL,
    tx_id = NULL,
    type = NULL,
    state = NULL,
    after = NULL,
    before = NULL,
    limit = NULL,
    config,
    tz = .okx_default_tz
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
wd_id	Character or 'NULL'. Withdrawal ID filter.
client_id	Character or 'NULL'. Client withdrawal ID filter.
tx_id	Character or 'NULL'. Transaction hash filter.
type	Character or 'NULL'. Withdrawal type filter.
state	Character or 'NULL'. Withdrawal state filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	API credentials as a list with api_key, secret_key, and passphrase.
tz	Timezone string (default: "Asia/Hong_Kong").

Value

A data.frame with withdrawal history rows and detailed transfer metadata.

get_copy_trade_config *Get copy trading config*

Description

Retrieve your account-level copy trading configuration.

Usage

```
get_copy_trade_config(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with top-level copy trading configuration fields. Nested 'details' are returned as a JSON string column.

get_copy_trade_current_subpos

Get current copy trading subpositions

Description

Retrieve your currently active subpositions under copy trading.

Usage

```
get_copy_trade_current_subpos(
  inst_type = NULL,
  inst_id = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
inst_id	Character or 'NULL'. Instrument ID filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Details

Wraps '/api/v5/copytrading/current-subpositions'. Returns one row per subposition, associated with the relevant lead trader.

Value

A 'data.frame' with fields like 'instId' and 'uniqueCode'.

Note

Since okxr 0.1.2

See Also

[get_copy_trade_historical_subpos()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_copy_trade_current_subpos(config = cfg)

## End(Not run)
```

get_copy_trade_historical_subpos

Get historical copy trading subpositions

Description

Retrieve your historical copy trading subpositions.

Usage

```
get_copy_trade_historical_subpos(
  inst_type = NULL,
  inst_id = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
inst_id	Character or 'NULL'. Instrument ID filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.

limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/copytrading/subpositions-history'. Returns one row per closed or historical subposition.

Value

A 'data.frame' with fields like 'instId' and 'uniqueCode'.

Note

Since okxr 0.1.2

See Also

[get_copy_trade_current_subpos()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
hist <- get_copy_trade_historical_subpos(config = cfg)
head(hist)

## End(Not run)
```

get_copy_trade_instruments

Get copy trading instruments

Description

Retrieve instruments currently available for copy trading.

Usage

```
get_copy_trade_instruments(inst_type = NULL, config, tz = .okx_default_tz)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with one row per instrument and an 'enabled' flag.

```
get_copy_trade_my_leaders
```

Get my lead traders

Description

Retrieve the list of lead traders you are currently copying.

Usage

```
get_copy_trade_my_leaders(
  inst_type = NULL,
  config,
  tz = .okx_default_tz,
  instType = inst_type
)
```

Arguments

inst_type	Character or 'NULL'. Filter by instrument type (e.g., "SWAP", "MARGIN", "SPOT"). If 'NULL', returns all.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
instType	Deprecated alias for 'inst_type'.

Details

Wraps '/api/v5/copytrading/current-lead-traders'. Returns one row per lead trader followed by your account.

Value

A 'data.frame' with fields such as 'nickName' and 'uniqueCode'.

Note

Since okxr 0.1.2

See Also

[get_copy_trade_settings()], [get_copy_trade_current_subpos()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_copy_trade_my_leaders(inst_type = "SWAP", config = cfg)

## End(Not run)
```

```
get_copy_trade_profit_sharing_details
      Get profit sharing details
```

Description

Retrieve realized profit sharing detail rows.

Usage

```
get_copy_trade_profit_sharing_details(
  inst_type = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "'Asia/Hong_Kong"'.

Value

A 'data.frame' with realized profit sharing rows.

```
get_copy_trade_public_config
```

Get public copy trading config

Description

Retrieve public copy trading limits and ratio bounds.

Usage

```
get_copy_trade_public_config(inst_type = NULL, tz = .okx_default_tz)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with public copy trading configuration fields.

```
get_copy_trade_public_copy_traders
```

Get public copy trader summary

Description

Retrieve public copy trader summary metrics for a lead trader.

Usage

```
get_copy_trade_public_copy_traders(
  unique_code,
  inst_type = NULL,
  limit = NULL,
  tz = .okx_default_tz
)
```

Arguments

unique_code	Character. Lead trader unique code.
inst_type	Character or 'NULL'. Instrument type filter.
limit	Integer or 'NULL'. Number of rows to request.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with summary metrics and a JSON-string 'copyTraders' column for nested trader details.

get_copy_trade_public_current_subpositions
Get public current copy trading subpositions

Description

Retrieve public current subpositions for a lead trader.

Usage

```
get_copy_trade_public_current_subpositions(  
  unique_code,  
  inst_type = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

unique_code	Character. Lead trader unique code.
inst_type	Character or 'NULL'. Instrument type filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with one row per current public subposition.

```
get_copy_trade_public_lead_traders
```

Get public lead trader ranks

Description

Retrieve ranked public lead trader summaries.

Usage

```
get_copy_trade_public_lead_traders(
    inst_type = NULL,
    sort_type = NULL,
    state = NULL,
    min_lead_days = NULL,
    min_assets = NULL,
    max_assets = NULL,
    min_aum = NULL,
    max_aum = NULL,
    data_ver = NULL,
    page = NULL,
    limit = NULL,
    tz = .okx_default_tz
)
```

Arguments

<code>inst_type</code>	Character or 'NULL'. Instrument type filter.
<code>sort_type</code>	Character or 'NULL'. Rank sort selector.
<code>state</code>	Character or 'NULL'. Lead trader state filter.
<code>min_lead_days</code>	Character or 'NULL'. Minimum lead-days selector.
<code>min_assets</code>	Character or 'NULL'. Minimum assets filter.
<code>max_assets</code>	Character or 'NULL'. Maximum assets filter.
<code>min_aum</code>	Character or 'NULL'. Minimum assets-under-management filter.
<code>max_aum</code>	Character or 'NULL'. Maximum assets-under-management filter.
<code>data_ver</code>	Character or 'NULL'. Data version selector used for pagination.
<code>page</code>	Character or 'NULL'. Page number.
<code>limit</code>	Integer or 'NULL'. Number of rows to request.
<code>tz</code>	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with top-level ranking metadata and a JSON-string 'ranks' column for nested leader rows.

```
get_copy_trade_public_pnl
```

Get public copy trading pnl

Description

Retrieve public pnl time windows for a lead trader.

Usage

```
get_copy_trade_public_pnl(  
    unique_code,  
    last_days,  
    inst_type = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

unique_code	Character. Lead trader unique code.
last_days	Character or numeric. OKX lookback selector.
inst_type	Character or 'NULL'. Instrument type filter.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with pnl windows and ratios.

```
get_copy_trade_public_preference_currency
```

Get public preference currencies

Description

Retrieve the most frequently traded currencies for a lead trader.

Usage

```
get_copy_trade_public_preference_currency(  
    unique_code,  
    inst_type = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

unique_code	Character. Lead trader unique code.
inst_type	Character or 'NULL'. Instrument type filter.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with preferred currencies and their ratios.

```
get_copy_trade_public_stats
```

Get public copy trading stats

Description

Retrieve public copy trading performance stats for a lead trader.

Usage

```
get_copy_trade_public_stats(
  unique_code,
  last_days,
  inst_type = NULL,
  tz = .okx_default_tz
)
```

Arguments

unique_code	Character. Lead trader unique code.
last_days	Character or numeric. OKX lookback selector.
inst_type	Character or 'NULL'. Instrument type filter.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with copy trading summary statistics.

`get_copy_trade_public_subpositions_history`*Get public historical copy trading subpositions*

Description

Retrieve public historical subpositions for a lead trader.

Usage

```
get_copy_trade_public_subpositions_history(  
    unique_code,  
    inst_type = NULL,  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

<code>unique_code</code>	Character. Lead trader unique code.
<code>inst_type</code>	Character or 'NULL'. Instrument type filter.
<code>after</code>	Character or 'NULL'. Pagination cursor for earlier records.
<code>before</code>	Character or 'NULL'. Pagination cursor for newer records.
<code>limit</code>	Integer or 'NULL'. Number of rows to request.
<code>tz</code>	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per historical public subposition.

`get_copy_trade_public_weekly_pnl`*Get public copy trading weekly pnl*

Description

Retrieve public weekly pnl series for a lead trader.

Usage

```
get_copy_trade_public_weekly_pnl(  
    unique_code,  
    inst_type = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

unique_code	Character. Lead trader unique code.
inst_type	Character or 'NULL'. Instrument type filter.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with weekly pnl windows and ratios.

get_copy_trade_settings
<i>Get copy trading settings</i>

Description

Retrieve your account's copy trading configuration.

Usage

```
get_copy_trade_settings(  
    unique_code,  
    inst_type = NULL,  
    config,  
    tz = .okx_default_tz,  
    uniqueCode = unique_code,  
    instType = inst_type  
)
```

Arguments

unique_code	Character. Lead trader unique code.
inst_type	Character or 'NULL'. Instrument type filter.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
uniqueCode	Deprecated alias for 'unique_code'.
instType	Deprecated alias for 'inst_type'.

Details

Wraps `"/api/v5/copytrading/copy-settings"`. Returns one row with the current copy mode and copy state for the given `'unique_code'`.

Value

A `'data.frame'` with fields like `'copyMode'` and `'copyState'`.

Note

Since okxr 0.1.2

See Also

[`get_copy_trade_my_leaders()`], [`get_copy_trade_current_subpos()`]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_copy_trade_settings(unique_code = "1129E65755274C36", config = cfg)

## End(Not run)
```

```
get_copy_trade_total_profit_sharing
  Get total profit sharing
```

Description

Retrieve total realized profit sharing by instrument type.

Usage

```
get_copy_trade_total_profit_sharing(
  inst_type = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

<code>inst_type</code>	Character or <code>'NULL'</code> . Instrument type filter.
<code>config</code>	List. API credentials/config.
<code>tz</code>	Character. Time zone for parsing timestamps. Default <code>"Asia/Hong_Kong"</code> .

Value

A 'data.frame' with total realized profit sharing rows.

```
get_copy_trade_total_unrealized_profit_sharing
```

Get total unrealized profit sharing

Description

Retrieve total unrealized profit sharing by instrument type.

Usage

```
get_copy_trade_total_unrealized_profit_sharing(  
  inst_type = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with total unrealized profit sharing rows.

```
get_copy_trade_unrealized_profit_sharing_details
```

Get unrealized profit sharing details

Description

Retrieve unrealized profit sharing detail rows.

Usage

```
get_copy_trade_unrealized_profit_sharing_details(  
  inst_type = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with unrealized profit sharing rows.

```
get_market_block_ticker
```

Get block ticker

Description

Retrieve the latest 24-hour block-trading volume for a single instrument.

Usage

```
get_market_block_ticker(inst_id, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A one-row 'data.frame' with block trading volume fields.

```
get_market_block_tickers
```

Get block tickers

Description

Retrieve the latest 24-hour block-trading volume for instruments under an instrument type.

Usage

```
get_market_block_tickers(
    inst_type,
    inst_family = NULL,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_type	Character. Instrument type.
inst_family	Character or 'NULL'. Instrument family filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per block ticker.

get_market_books	<i>Get order book</i>
------------------	-----------------------

Description

Retrieve the current order book for an instrument.

Usage

```
get_market_books(inst_id, sz = NULL, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT".
sz	Integer or 'NULL'. Order book depth. If 'NULL', OKX uses its endpoint default.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with JSON-encoded 'asks' and 'bids' columns plus 'ts'.

get_market_books_rpi *Get RPI order book*

Description

Retrieve the RPI order book for an instrument.

Usage

```
get_market_books_rpi(inst_id, sz = NULL, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT-SWAP".
sz	Integer or 'NULL'. Order book depth. If 'NULL', OKX uses its endpoint default.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with JSON-encoded 'asks' and 'bids' columns plus timestamp and sequence meta-data when returned by OKX.

get_market_candles *Get recent market candles*

Description

Retrieve the latest candlestick data for a given instrument and bar size.

Usage

```
get_market_candles(  
  inst_id,  
  bar,  
  limit = 100L,  
  config = NULL,  
  tz = .okx_default_tz,  
  standardize_names = TRUE  
)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT", "ETH-USDT-SWAP".
bar	Character. Candlestick granularity, e.g. "1m", "5m", "1H", "1D".
limit	Integer. Number of bars to retrieve. Default '100L'.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'. standardize_names
	Logical. If 'TRUE' (default), renames columns to 'timestamp', 'open', 'high', 'low', 'close', 'volume', 'volQuote'.

Details

Wraps '/api/v5/market/candles'. Returns up to 'limit' bars, sorted by timestamp. Candlestick fields can be standardized to common OHLCV names via 'standardize_names = TRUE'.

Value

A 'data.frame' with columns including 'timestamp', 'open', 'high', 'low', 'close', 'volume', and 'volQuote'. Timestamps are 'POSIXct' in 'tz'.

Note

Since okxr 0.1.1

See Also

[get_market_history_candles()], [get_public_mark_price()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_market_candles("BTC-USDT", bar = "5m", limit = 50, config = cfg)

## End(Not run)
```

```
get_market_exchange_rate
```

Get exchange rate

Description

Retrieve the two-week average exchange rate series summary.

Usage

```
get_market_exchange_rate(config = NULL, tz = .okx_default_tz)
```

Arguments

config	Optional list. Public endpoint request options, such as ‘timeout’; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default “Asia/Hong_Kong”.

Value

A one-row ‘data.frame’ with ‘usdCny’.

```
get_market_history_candles
```

Get historical market candles

Description

Retrieve candlestick data before a specific datetime.

Usage

```
get_market_history_candles(
  inst_id,
  bar,
  before = NULL,
  limit = 100L,
  config = NULL,
  tz = .okx_default_tz,
  standardize_names = TRUE
)
```

Arguments

inst_id	Character. Instrument ID, e.g. “BTC-USDT”.
bar	Character. Candlestick granularity, e.g. “1m”, “5m”, “1H”.
before	Character or ‘NULL’. Timestamp string like “%Y-%m-%d %H:%M:%S”. If ‘NULL’, fetch recent history.
limit	Integer. Number of bars to retrieve. Default ‘100L’.
config	Optional list. Public endpoint request options, such as ‘timeout’; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default “Asia/Hong_Kong”.
standardize_names	Logical. If ‘TRUE’ (default), renames columns to ‘timestamp’, ‘open’, ‘high’, ‘low’, ‘close’, ‘volume’, ‘volQuote’.

Details

Wraps `/api/v5/market/history-candles`. If `'before'` is supplied, it is converted to milliseconds since epoch (in `'tz'`) and sent as `'after=...'` (per OKX semantics: **return data before this time**).

Value

A `'data.frame'` of candlestick bars. If `'standardize_names = TRUE'`, column names are normalized. Timestamps are `'POSIXct'` in `'tz'`.

Note

Since okxr 0.1.1

See Also

[`get_market_candles()`]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_market_history_candles(
  "ETH-USDT-SWAP", bar = "1H",
  before = "2025-08-20 00:00:00", config = cfg
)

## End(Not run)
```

`get_market_history_index_candles`
Get historical index candles

Description

Retrieve historical candlestick data for an index.

Usage

```
get_market_history_index_candles(
  inst_id,
  bar = NULL,
  after = NULL,
  before = NULL,
  limit = 100L,
  config = NULL,
  tz = .okx_default_tz,
  standardize_names = TRUE
)
```

Arguments

inst_id	Character. Index ID.
bar	Character or 'NULL'. Bar size.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
limit	Integer. Number of rows to request. Default '100L'.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
standardize_names	Logical. If 'TRUE' (default), renames OHLC columns to 'timestamp', 'open', 'high', 'low', and 'close'.

Value

A 'data.frame' of historical index candlesticks.

```
get_market_history_mark_price_candles
```

Get historical mark price candles

Description

Retrieve historical candlestick data for mark price.

Usage

```
get_market_history_mark_price_candles(
  inst_id,
  bar = NULL,
  after = NULL,
  before = NULL,
  limit = 100L,
  config = NULL,
  tz = .okx_default_tz,
  standardize_names = TRUE
)
```

Arguments

inst_id	Character. Instrument ID.
bar	Character or 'NULL'. Bar size.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.

limit	Integer. Number of rows to request. Default '100L'.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
standardize_names	Logical. If 'TRUE' (default), renames OHLC columns to 'timestamp', 'open', 'high', 'low', and 'close'.

Value

A 'data.frame' of historical mark-price candlesticks.

get_market_history_trades	<i>Get historical public trades</i>
---------------------------	-------------------------------------

Description

Retrieve public trade history for an instrument.

Usage

```
get_market_history_trades(
  inst_id,
  type = NULL,
  after = NULL,
  before = NULL,
  limit = NULL,
  config = NULL,
  tz = .okx_default_tz
)
```

Arguments

inst_id	Character. Instrument ID, e.g. '"BTC-USDT"'.
type	Character or 'NULL'. Pagination type, using OKX values.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with historical public trades.

`get_market_index_candles`*Get recent index candles*

Description

Retrieve the latest candlestick data for an index.

Usage

```
get_market_index_candles(  
    inst_id,  
    bar = NULL,  
    after = NULL,  
    before = NULL,  
    limit = 100L,  
    config = NULL,  
    tz = .okx_default_tz,  
    standardize_names = TRUE  
)
```

Arguments

<code>inst_id</code>	Character. Index ID.
<code>bar</code>	Character or 'NULL'. Bar size.
<code>after</code>	Character or 'NULL'. Pagination cursor for older rows.
<code>before</code>	Character or 'NULL'. Pagination cursor for newer rows.
<code>limit</code>	Integer. Number of rows to request. Default '100L'.
<code>config</code>	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
<code>tz</code>	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
<code>standardize_names</code>	Logical. If 'TRUE' (default), renames OHLC columns to 'timestamp', 'open', 'high', 'low', and 'close'.

Value

A 'data.frame' of index candlesticks.

```
get_market_index_components
```

Get index components

Description

Retrieve component-exchange information for an index.

Usage

```
get_market_index_components(index, config = NULL, tz = .okx_default_tz)
```

Arguments

index	Character. Index identifier, e.g. "BTC-USD".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A one-row 'data.frame' with 'index', 'last', 'ts', and JSON-encoded 'components'.

```
get_market_index_tickers
```

Get index tickers

Description

Retrieve the latest public index-price snapshots.

Usage

```
get_market_index_tickers(
  quote_ccy = NULL,
  inst_id = NULL,
  config = NULL,
  tz = .okx_default_tz
)
```

Arguments

quote_ccy	Character or 'NULL'. Quote currency filter.
inst_id	Character or 'NULL'. Specific index ID filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per index ticker.

get_market_mark_price_candles
<i>Get mark price candles</i>

Description

Retrieve recent candlestick data for the mark price of an instrument.

Usage

```
get_market_mark_price_candles(  
  inst_id,  
  bar = NULL,  
  after = NULL,  
  before = NULL,  
  limit = 100L,  
  config = NULL,  
  tz = .okx_default_tz,  
  standardize_names = TRUE  
)
```

Arguments

inst_id	Character. Instrument ID.
bar	Character or 'NULL'. Bar size.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
limit	Integer. Number of rows to request. Default '100L'.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.
standardize_names	Logical. If 'TRUE' (default), renames OHLC columns to 'timestamp', 'open', 'high', 'low', and 'close'.

Value

A 'data.frame' of mark-price candlesticks.

```
get_market_option_instrument_family_trades
```

Get option trades by instrument family

Description

Retrieve recent option trades for all instruments under the same instrument family.

Usage

```
get_market_option_instrument_family_trades(
    inst_family,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

<code>inst_family</code>	Character. Instrument family, e.g. "BTC-USD".
<code>config</code>	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
<code>tz</code>	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with recent option trades under the requested instrument family, including 'instId', 'tradeId', 'px', 'sz', 'side', and 'ts'.

```
get_market_platform_24_volume
```

Get platform 24-hour volume

Description

Retrieve total platform order-book trading volume over the last 24 hours.

Usage

```
get_market_platform_24_volume(config = NULL, tz = .okx_default_tz)
```

Arguments

<code>config</code>	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
<code>tz</code>	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A one-row 'data.frame' with 'volUsd', 'volCny', and 'ts'.

get_market_ticker	<i>Get market ticker</i>
-------------------	--------------------------

Description

Retrieve the latest ticker snapshot for a specific instrument.

Usage

```
get_market_ticker(inst_id, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT" or "ETH-USDT-SWAP".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with the latest ticker fields returned by OKX.

get_market_tickers	<i>Get market tickers</i>
--------------------	---------------------------

Description

Retrieve ticker snapshots for all instruments under an instrument type.

Usage

```
get_market_tickers(  
  inst_type,  
  uly = NULL,  
  inst_family = NULL,  
  config = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type, e.g. "SPOT", "SWAP", "FUTURES", or "OPTION".
uly	Character or 'NULL'. Underlying. Optional filter for derivatives.
inst_family	Character or 'NULL'. Instrument family. Optional filter for derivatives and options.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with one row per ticker.

get_market_trades	<i>Get recent public trades</i>
-------------------	---------------------------------

Description

Retrieve recent public trades for an instrument.

Usage

```
get_market_trades(inst_id, limit = NULL, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT".
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with recent public trades.

`get_public_block_trades`*Get public block trades*

Description

Retrieve recent single-leg public block trades for an instrument.

Usage

```
get_public_block_trades(inst_id, config = NULL, tz = .okx_default_tz)
```

Arguments

<code>inst_id</code>	Character. Instrument ID.
<code>config</code>	Optional list. Public endpoint request options, such as ‘timeout’; credentials are not required.
<code>tz</code>	Character. Time zone for parsing timestamps. Default “Asia/Hong_Kong”.

Value

A ‘data.frame’ with block trade fields such as price, size, trade side, volatility, and timestamps.

`get_public_convert_contract_coin`*Convert between contract size and currency amount*

Description

Convert the crypto value to the number of contracts, or vice versa.

Usage

```
get_public_convert_contract_coin(  
  inst_id,  
  sz,  
  type = NULL,  
  px = NULL,  
  unit = NULL,  
  op_type = NULL,  
  config = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

inst_id	Character. Instrument ID.
sz	Character or numeric. Quantity to convert.
type	Character or 'NULL'. Convert type: "1" for currency to contract, "2" for contract to currency.
px	Character, numeric, or 'NULL'. Optional order price.
unit	Character or 'NULL'. Currency unit, "coin" or "usds".
op_type	Character or 'NULL'. Order type for futures or swaps, such as "open" or "close".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with conversion result fields 'type', 'instId', 'px', 'sz', and 'unit'.

```
get_public_delivery_exercise_history
```

Get delivery or exercise history

Description

Retrieve futures delivery records or option exercise records.

Usage

```
get_public_delivery_exercise_history(
  inst_type,
  inst_family,
  after = NULL,
  before = NULL,
  limit = NULL,
  config = NULL,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character. "FUTURES" or "OPTION".
inst_family	Character. Instrument family.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.

limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with 'ts' and JSON-encoded 'details'.

get_public_discount_rate_interest_free_quota

Get collateral discount rate and interest-free quota

Description

Retrieve public collateral discount-rate tiers and interest-free quota information for supported currencies.

Usage

```
get_public_discount_rate_interest_free_quota(
  ccy = NULL,
  discount_lv = NULL,
  config = NULL,
  tz = .okx_default_tz
)
```

Arguments

ccy	Character or 'NULL'. Currency filter, e.g. '"BTC"'.
discount_lv	Character or 'NULL'. Discount level filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with top-level discount and interest-free quota fields; nested tier details are JSON-encoded in 'details'.

get_public_economic_calendar

Get economic calendar

Description

Retrieve macro-economic calendar records. OKX requires authentication for this endpoint.

Usage

```
get_public_economic_calendar(
    region = NULL,
    importance = NULL,
    before = NULL,
    after = NULL,
    limit = NULL,
    config,
    tz = .okx_default_tz
)
```

Arguments

region	Character or 'NULL'. Region filter.
importance	Character or 'NULL'. Importance level filter.
before	Character or 'NULL'. Pagination cursor for newer rows.
after	Character or 'NULL'. Pagination cursor for older rows.
limit	Integer or 'NULL'. Number of rows to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with calendar event fields and timestamps.

get_public_estimated_price

Get estimated delivery or exercise price

Description

Retrieve the estimated delivery, exercise, or settlement price for derivatives and events instruments.

Usage

```
get_public_estimated_price(
    inst_type,
    inst_family = NULL,
    inst_id = NULL,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_type	Character. Instrument type, such as "FUTURES", "OPTION", "SWAP", or "EVENTS".
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Specific instrument ID filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instType', 'instId', 'settlePx', and 'ts'.

```
get_public_estimated_settlement_info
```

Get estimated settlement info

Description

Retrieve the estimated settlement price for a futures instrument close to settlement.

Usage

```
get_public_estimated_settlement_info(
    inst_id,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_id	Character. Instrument ID.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'nextSettleTime', 'estSettlePx', and 'ts'.

get_public_event_contract_events
<i>Get event-contract events</i>

Description

Retrieve event records for a prediction-market series.

Usage

```
get_public_event_contract_events(  
  series_id,  
  event_id = NULL,  
  state = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  config = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

series_id	Character. Series ID.
event_id	Character or 'NULL'. Event ID filter.
state	Character or 'NULL'. Event state filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with event identifiers, times, and state.

```
get_public_event_contract_markets
```

Get event-contract markets

Description

Retrieve market records for prediction-market events.

Usage

```
get_public_event_contract_markets(  
    series_id,  
    event_id = NULL,  
    inst_id = NULL,  
    state = NULL,  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

series_id	Character. Series ID.
event_id	Character or 'NULL'. Event ID filter.
inst_id	Character or 'NULL'. Instrument ID filter.
state	Character or 'NULL'. Market state filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options.
tz	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with market identifiers, strike metadata, state, and settlement fields.

```
get_public_event_contract_series
```

Get event-contract series

Description

Retrieve prediction-market series metadata.

Usage

```
get_public_event_contract_series(
    series_id = NULL,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

series_id	Character or 'NULL'. Series ID filter. If 'NULL', OKX returns all series.
config	Optional list. Public endpoint request options.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with series metadata. Nested settlement information is JSON-encoded in 'settlement'.

```
get_public_funding_rate
```

Get current funding rate

Description

Retrieve the current funding rate for a perpetual swap instrument.

Usage

```
get_public_funding_rate(inst_id, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT-SWAP".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' containing the current funding rate fields returned by OKX.

```
get_public_funding_rate_history
```

Get funding rate history

Description

Retrieve historical funding rate entries for a perpetual swap instrument.

Usage

```
get_public_funding_rate_history(  
  inst_id,  
  before = NULL,  
  after = NULL,  
  limit = 400,  
  config = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT-SWAP".
before	Optional cursor for records earlier than the supplied value.
after	Optional cursor for records later than the supplied value.
limit	Integer. Number of records to request. Default '400'.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' containing funding rate history rows returned by OKX.

get_public_instruments

Get instrument metadata

Description

Retrieve metadata for instruments of a given type.

Usage

```
get_public_instruments(
    inst_id = NULL,
    inst_type = "SWAP",
    series_id = NULL,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_id	Character or 'NULL'. Specific instrument ID to query. Use 'NULL' to fetch all instruments of 'inst_type'.
inst_type	Character. Instrument type. One of "SPOT", "MARGIN", "SWAP" (default), "FUTURES", "OPTION", or "EVENTS".
series_id	Character or 'NULL'. Event-contract series ID. Required by OKX when 'inst_type' = "EVENTS".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/public/instruments'. Returns one row per instrument, including contract specifications, tick size, lot size, expiry, and state.

Value

A 'data.frame' with instrument metadata (e.g., 'instType', 'instId', 'uly', 'baseCcy', 'quoteCcy', 'settleCcy', 'ctVal', 'ctMult', 'tickSz', 'lotSz', 'minSz', 'expTime', 'lever', 'state', ...).

Note

Since okxr 0.1.2

See Also

[get_public_mark_price()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
# Get metadata for all SWAP instruments
df <- get_public_instruments(inst_type = "SWAP", config = cfg)

# Get metadata for one instrument
get_public_instruments("ETH-USDT-SWAP", inst_type = "SWAP", config = cfg)

## End(Not run)
```

```
get_public_instrument_tick_bands
```

Get option instrument tick bands

Description

Retrieve option tick-band information for one or more option instrument families.

Usage

```
get_public_instrument_tick_bands(
  inst_type = "OPTION",
  inst_family = NULL,
  config = NULL,
  tz = .okx_default_tz
)
```

Arguments

inst_type	Character. Instrument type, "OPTION" or "EVENTS".
inst_family	Character or 'NULL'. Instrument family filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instType', 'instFamily', and JSON-encoded 'tickBand' details.

get_public_insurance_fund
<i>Get security fund balance information</i>

Description

Retrieve public insurance-fund or security-fund balance information.

Usage

```
get_public_insurance_fund(  
    inst_type,  
    type = NULL,  
    inst_family = NULL,  
    ccy = NULL,  
    before = NULL,  
    after = NULL,  
    limit = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type, such as "MARGIN", "SWAP", "FUTURES", or "OPTION".
type	Character or 'NULL'. Fund update type filter.
inst_family	Character or 'NULL'. Instrument family filter for derivatives.
ccy	Character or 'NULL'. Currency filter for margin data.
before	Character or 'NULL'. Pagination cursor for newer rows.
after	Character or 'NULL'. Pagination cursor for older rows.
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with top-level fund totals; nested per-currency detail rows are JSON-encoded in 'details'.

```
get_public_interest_rate_loan_quota
```

Get interest rate and loan quota

Description

Retrieve public borrowing-rate and loan-quota tables.

Usage

```
get_public_interest_rate_loan_quota(  
    ccy = NULL,  
    vip_level = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

ccy	Character or 'NULL'. Currency filter.
vip_level	Character or 'NULL'. VIP level filter when supported by OKX.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "'Asia/Hong_Kong"'.

Value

A 'data.frame' where nested basic, VIP, regular, and custom quota tables are JSON-encoded string columns.

```
get_public_mark_price
```

Get current mark price

Description

Retrieve the current mark price for a given instrument.

Usage

```
get_public_mark_price(  
    inst_id,  
    inst_type = "SWAP",  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT", "ETH-USDT-SWAP".
inst_type	Character. Instrument type. One of "SPOT", "MARGIN", "SWAP" (default), "FUTURES", "OPTION".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/public/mark-price'. Useful for margin calculations and PnL estimation. Returns a single row with the latest mark price and timestamp.

Value

A 'data.frame' with columns 'timestamp', 'instId', 'markPx'.

Note

Since okxr 0.1.1

See Also

[get_public_instruments()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_public_mark_price("BTC-USDT", inst_type = "SWAP", config = cfg)

## End(Not run)
```

```
get_public_mm_instrument_types
```

Get MM instrument types

Description

Retrieve MM Program instrument type classifications for SPOT and SWAP instruments.

Usage

```
get_public_mm_instrument_types(
    inst_type = NULL,
    inst_id = NULL,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter, "SPOT" or "SWAP". If 'NULL', OKX returns all supported types.
inst_id	Character or 'NULL'. Instrument ID filter. If supplied, OKX returns at most one record.
config	Optional list. Public endpoint request options.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'instType', and 'pairType'.

```
get_public_open_interest
```

Get open interest

Description

Retrieve current open interest for an instrument.

Usage

```
get_public_open_interest(
    inst_id,
    inst_type,
    config = NULL,
    tz = .okx_default_tz
)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT-SWAP".
inst_type	Character. Instrument type such as "SWAP" or "FUTURES".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' containing open interest fields returned by OKX.

get_public_option_trades
<i>Get public option trades</i>

Description

Retrieve recent public option trades filtered by instrument ID or instrument family.

Usage

```
get_public_option_trades(  
  inst_id = NULL,  
  inst_family = NULL,  
  opt_type = NULL,  
  config = NULL,  
  tz = .okx_default_tz  
)
```

Arguments

inst_id	Character or 'NULL'. Specific option instrument ID.
inst_family	Character or 'NULL'. Option instrument family, e.g. "BTC-USD". Either 'inst_id' or 'inst_family' should be supplied.
opt_type	Character or 'NULL'. Option type filter: "C" for call or "P" for put.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with recent option trade rows, including option instrument identifiers, trade price and size, option side/type, forward, index and mark prices, implied volatility, and trade time.

`get_public_opt_summary`*Get option summary*

Description

Retrieve option market summary data for an instrument family.

Usage

```
get_public_opt_summary(  
    inst_family,  
    exp_time = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

<code>inst_family</code>	Character. Option instrument family.
<code>exp_time</code>	Character or 'NULL'. Expiry date in 'YYMMDD'.
<code>config</code>	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
<code>tz</code>	Character. Time zone for parsing timestamps. Default '"Asia/Hong_Kong"'.

Value

A 'data.frame' with option greeks, volatility, forward price, and timestamp fields.

`get_public_position_tiers`*Get public position tiers*

Description

Retrieve public tier, margin, and maximum leverage information.

Usage

```
get_public_position_tiers(  
    inst_type,  
    td_mode,  
    inst_family = NULL,  
    inst_id = NULL,  
    ccy = NULL,  
    tier = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type.
td_mode	Character. Trade mode.
inst_family	Character or 'NULL'. Instrument family.
inst_id	Character or 'NULL'. Instrument ID(s).
ccy	Character or 'NULL'. Margin currency.
tier	Character or 'NULL'. Tier filter.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "'Asia/Hong_Kong'".

Value

A 'data.frame' with public position tier information.

get_public_premium_history
<i>Get premium history</i>

Description

Retrieve premium-index history for an instrument.

Usage

```
get_public_premium_history(  
    inst_id,  
    after = NULL,  
    before = NULL,  
    bar = NULL,  
    limit = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

inst_id	Character. Instrument ID.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
bar	Character or 'NULL'. Bar size such as "1m" or "1H".
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'instId', 'premium', and 'ts'.

`get_public_price_limit`*Get price limit*

Description

Retrieve buy and sell price limits for an instrument.

Usage

```
get_public_price_limit(inst_id, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_id	Character. Instrument ID, e.g. "BTC-USDT-SWAP".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with price limit fields.

get_public_settlement_history	<i>Get settlement history</i>
-------------------------------	-------------------------------

Description

Retrieve futures settlement history for an instrument family.

Usage

```
get_public_settlement_history(  
    inst_family,  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config = NULL,  
    tz = .okx_default_tz  
)
```

Arguments

inst_family	Character. Instrument family.
after	Character or 'NULL'. Pagination cursor for older rows.
before	Character or 'NULL'. Pagination cursor for newer rows.
limit	Integer or 'NULL'. Number of rows to request.
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with 'ts' and JSON-encoded 'details'.

get_public_time	<i>Get OKX system time</i>
-----------------	----------------------------

Description

Retrieve OKX system time.

Usage

```
get_public_time(config = NULL, tz = .okx_default_tz)
```

Arguments

config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A one-row 'data.frame' with system time.

get_public_underlying *Get underlying list*

Description

Retrieve available underlyings for derivatives instruments.

Usage

```
get_public_underlying(inst_type, config = NULL, tz = .okx_default_tz)
```

Arguments

inst_type	Character. Instrument type: "SWAP", "FUTURES", or "OPTION".
config	Optional list. Public endpoint request options, such as 'timeout'; credentials are not required.
tz	Character. Unused except for interface consistency.

Value

A one-column 'data.frame' with 'uly'.

get_trade_account_rate_limit
Get trade account rate limit

Description

Retrieve account rate limit information related to new and amended order requests.

Usage

```
get_trade_account_rate_limit(config, tz = .okx_default_tz)
```

Arguments

- config List. API credentials/config.
- tz Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

Wraps '/api/v5/trade/account-rate-limit'. Returns one row containing the current account rate limit and fill-ratio metrics used by OKX.

Value

A one-row 'data.frame' with rate-limit metrics such as 'accRateLimit', 'fillRatio', 'mainFillRatio', 'nextAccRateLimit', and 'ts'.

get_trade_easy_convert_currency_list
<i>Get easy convert currency list</i>

Description

Retrieve currencies available for easy convert.

Usage

```
get_trade_easy_convert_currency_list(  
  source = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

- source Character or 'NULL'. Source account type filter.
- config List. API credentials/config.
- tz Character. Time zone for parsing timestamps.

Value

A 'data.frame' with source and target currency metadata.

get_trade_easy_convert_history	<i>Get easy convert history</i>
--------------------------------	---------------------------------

Description

Retrieve easy convert history.

Usage

```
get_trade_easy_convert_history(  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with easy convert history rows.

get_trade_fills	<i>Get trade fills</i>
-----------------	------------------------

Description

Retrieve recently filled transaction details from the last 3 days.

Usage

```
get_trade_fills(  
  inst_type = NULL,  
  inst_family = NULL,  
  inst_id = NULL,  
  ord_id = NULL,  
  sub_type = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  begin = NULL,  
  end = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character or 'NULL'. Instrument type filter.
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Instrument ID filter.
ord_id	Character or 'NULL'. Order ID filter.
sub_type	Character or 'NULL'. Transaction subtype filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
begin	Character or 'NULL'. Begin timestamp in milliseconds.
end	Character or 'NULL'. End timestamp in milliseconds.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "'Asia/Hong_Kong'".

Value

A 'data.frame' with fill rows.

get_trade_fills_history
<i>Get trade fills history</i>

Description

Retrieve historical filled transaction details from the last 3 months.

Usage

```

get_trade_fills_history(
    inst_type,
    inst_family = NULL,
    inst_id = NULL,
    ord_id = NULL,
    sub_type = NULL,
    after = NULL,
    before = NULL,
    limit = NULL,
    begin = NULL,
    end = NULL,
    config,
    tz = .okx_default_tz
)

```

Arguments

inst_type	Character. Instrument type, e.g. "SPOT" or "SWAP".
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Instrument ID filter.
ord_id	Character or 'NULL'. Order ID filter.
sub_type	Character or 'NULL'. Transaction subtype filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of rows to request.
begin	Character or 'NULL'. Begin timestamp in milliseconds.
end	Character or 'NULL'. End timestamp in milliseconds.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with historical fill rows.

```

get_trade_one_click_repay_currency_list
  Get one-click repay currency list

```

Description

Retrieve repayable currencies for the legacy one-click repay endpoint.

Usage

```
get_trade_one_click_repay_currency_list(  
    debt_type = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

debt_type	Character or 'NULL'. Debt type filter.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with debt and repay currency metadata.

```
get_trade_one_click_repay_currency_list_v2
```

Get one-click repay currency list v2

Description

Retrieve repayable currencies for the new one-click repay endpoint.

Usage

```
get_trade_one_click_repay_currency_list_v2(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with debt and repay currency metadata.

```
get_trade_one_click_repay_history
```

Get one-click repay history

Description

Retrieve legacy one-click repay history.

Usage

```
get_trade_one_click_repay_history(  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with one-click repay history rows.

```
get_trade_one_click_repay_history_v2
```

Get one-click repay history v2

Description

Retrieve new one-click repay history.

Usage

```
get_trade_one_click_repay_history_v2(  
    after = NULL,  
    before = NULL,  
    limit = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with one-click repay history rows.

get_trade_order	<i>Get trade order details</i>
-----------------	--------------------------------

Description

Retrieve detailed information about a specific OKX order by either the exchange-assigned 'ord_id' or your client-assigned 'cl_ord_id'.

Usage

```
get_trade_order(
  inst_id,
  ord_id = NULL,
  cl_ord_id = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_id	Character. Instrument ID, e.g. "ETH-USDT-SWAP" (perps), "BTC-USDT" (spot), or "BTC-USD-240927" (dated futures).
ord_id	Character, optional. The OKX order ID. Provide this or 'cl_ord_id'.
cl_ord_id	Character, optional. Your client order ID. Provide this or 'ord_id'.
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'. May also include 'base_url'.
tz	Character. Time zone for parsing timestamps. Default "Asia/Hong_Kong".

Details

You must provide exactly one identifier: 'ord_id' **or** 'cl_ord_id'. Timestamps in the response are converted to 'POSIXct' in the supplied 'tz'.

Value

A 'data.frame' (one row) with order details following OKX schema (e.g., 'ordId', 'clOrdId', 'instId', 'ordType', 'px', 'sz', 'side', 'posSide', 'tdMode', 'accFillSz', 'fillPx', 'fillSz', 'fillTime', 'avgPx', 'state', 'lever', etc.). Timestamp columns are 'POSIXct' in 'tz'.

Common errors

- 'Either 'ord_id' or 'cl_ord_id' must be provided.' (client-side) - HTTP 401 Unauthorized (missing/invalid credentials) - OKX 'code' like '51000' invalid sign or '51603' order not found

Note

Since okxr 0.1.1

See Also

[get_trade_orders_pending()], [get_trade_orders_history_7d()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
get_trade_order(
  inst_id = "ETH-USDT-SWAP",
  ord_id = "1234567890",
  config = cfg
)

## End(Not run)
```

get_trade_orders_algo_history

Get algo order history

Description

Retrieve historical algo orders.

Usage

```
get_trade_orders_algo_history(
  ord_type,
  state = NULL,
  algo_id = NULL,
  inst_type = NULL,
  inst_id = NULL,
  after = NULL,
```

```
before = NULL,  
limit = NULL,  
config,  
tz = .okx_default_tz  
)
```

Arguments

ord_type	Character. Algo order type filter.
state	Character or 'NULL'. Algo order state filter.
algo_id	Character or 'NULL'. Algo order ID.
inst_type	Character or 'NULL'. Instrument type filter.
inst_id	Character or 'NULL'. Instrument ID filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with algo order history rows.

get_trade_orders_algo_pending
<i>Get pending algo orders</i>

Description

Retrieve untriggered algo orders.

Usage

```
get_trade_orders_algo_pending(  
  ord_type,  
  algo_id = NULL,  
  inst_type = NULL,  
  inst_id = NULL,  
  after = NULL,  
  before = NULL,  
  limit = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

ord_type	Character. Algo order type filter.
algo_id	Character or 'NULL'. Algo order ID.
inst_type	Character or 'NULL'. Instrument type filter.
inst_id	Character or 'NULL'. Instrument ID filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with pending algo orders.

get_trade_orders_history
<i>Get trade orders history</i>

Description

Retrieve completed orders from the last 7 days.

Usage

```
get_trade_orders_history(  
  inst_type,  
  inst_family = NULL,  
  inst_id = NULL,  
  ord_type = NULL,  
  state = NULL,  
  category = NULL,  
  after = NULL,  
  before = NULL,  
  begin = NULL,  
  end = NULL,  
  limit = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type.
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Instrument ID filter.
ord_type	Character or 'NULL'. Order type filter.
state	Character or 'NULL'. Order state filter.
category	Character or 'NULL'. Order category filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
begin	Character or 'NULL'. Begin timestamp in milliseconds.
end	Character or 'NULL'. End timestamp in milliseconds.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with historical order rows.

get_trade_orders_history_7d
Get trade orders history (last 7 days)

Description

Retrieve recent order history for an instrument type.

Usage

```
get_trade_orders_history_7d(inst_type = "SWAP", config, tz = .okx_default_tz)
```

Arguments

inst_type	Character. Instrument type. One of "SPOT", "MARGIN", "SWAP", "FUTURES", "OPTION". Default "SWAP".
config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'. May also include 'base_url'.
tz	Character. Time zone for parsing timestamps (e.g. "Asia/Hong_Kong").

Details

This wraps '/api/v5/trade/orders-history' and covers about 7 days of data. Older data is available from OKX's archive endpoint.

Value

A 'data.frame' with one row per historical order and columns following the OKX schema (same layout as pending orders, plus final states). Timestamp columns are 'POSIXct' in 'tz'.

Common errors

- HTTP 401 Unauthorized - HTTP 400 Bad Request for invalid 'inst_type'

Note

Since okxr 0.1.2

See Also

[get_trade_order()], [get_trade_orders_pending()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
hist <- get_trade_orders_history_7d(
  inst_type = "SWAP",
  config = cfg,
  tz = "Asia/Hong_Kong"
)
tail(hist)

## End(Not run)
```

get_trade_orders_history_archive

Get archived trade orders history

Description

Retrieve completed orders from the last 3 months.

Usage

```
get_trade_orders_history_archive(
  inst_type,
  inst_family = NULL,
  inst_id = NULL,
  ord_type = NULL,
  state = NULL,
  category = NULL,
  after = NULL,
```

```
before = NULL,  
begin = NULL,  
end = NULL,  
limit = NULL,  
config,  
tz = .okx_default_tz  
)
```

Arguments

inst_type	Character. Instrument type.
inst_family	Character or 'NULL'. Instrument family filter.
inst_id	Character or 'NULL'. Instrument ID filter.
ord_type	Character or 'NULL'. Order type filter.
state	Character or 'NULL'. Order state filter.
category	Character or 'NULL'. Order category filter.
after	Character or 'NULL'. Pagination cursor for earlier records.
before	Character or 'NULL'. Pagination cursor for newer records.
begin	Character or 'NULL'. Begin timestamp in milliseconds.
end	Character or 'NULL'. End timestamp in milliseconds.
limit	Integer or 'NULL'. Number of results to request.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with archived order rows.

get_trade_orders_pending
<i>Get all pending trade orders</i>

Description

Retrieve all currently open (unfilled) orders for your OKX account.

Usage

```
get_trade_orders_pending(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config, typically containing 'api_key', 'secret_key', and 'passphrase'. May also include 'base_url'.
tz	Character. Time zone for parsing timestamps (e.g. "Asia/Hong_Kong").

Details

Returns one row per open order. Timestamps are parsed to 'POSIXct' using 'tz'.

Value

A 'data.frame' with one row per pending order and columns following the OKX schema (e.g., 'cTime', 'ordId', 'clOrdId', 'tag', 'instId', 'ordType', 'px', 'sz', 'side', 'posSide', 'tdMode', 'accFillSz', 'fillPx', 'fillSz', 'fillTime', 'avgPx', 'state', 'lever', ...). Timestamp columns are 'POSIXct'.

Common errors

- HTTP 401 Unauthorized (missing/invalid credentials) - Rate limiting: HTTP 429 / OKX throttle codes

Note

Since okxr 0.1.1

See Also

[get_trade_order()], [get_trade_orders_history_7d()]

Examples

```
## Not run:
cfg <- list(api_key = "xxx", secret_key = "xxx", passphrase = "xxx")
df <- get_trade_orders_pending(config = cfg, tz = "Asia/Hong_Kong")
head(df)

## End(Not run)
```

get_trade_order_algo *Get a single algo order*

Description

Retrieve a specific algo order by algo ID or client algo order ID.

Usage

```
get_trade_order_algo(
  algo_id = NULL,
  algo_cl_ord_id = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

algo_id	Character or 'NULL'. Algo order ID.
algo_cl_ord_id	Character or 'NULL'. Client algo order ID.
config	List. API credentials/config.
tz	Character. Time zone for parsing timestamps.

Value

A 'data.frame' with algo order details.

get_user_glp_historical_performance
<i>Get historical GLP performance</i>

Description

Retrieve daily Global Liquidity Program (GLP) performance for one program.

Usage

```
get_user_glp_historical_performance(
  program,
  begin = NULL,
  end = NULL,
  limit = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

program	Character. GLP program: "SPOT", "PERP", or "FUT_NTO".
begin	Character or numeric. Optional inclusive Unix time in milliseconds.
end	Character or numeric. Optional inclusive Unix time in milliseconds.
limit	Integer or 'NULL'. Optional maximum number of rows, up to 100.
config	List. API credentials/config with read permission.
tz	Character. Time zone for parsing response timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with the GLP performance date. Nested 'volume' and 'share' objects are retained as JSON strings.

```
get_user_glp_today_performance
```

Get GLP performance for today

Description

Retrieve the current-day and month-to-date Global Liquidity Program (GLP) performance snapshot for the authenticated market-maker account.

Usage

```
get_user_glp_today_performance(config, tz = .okx_default_tz)
```

Arguments

config	List. API credentials/config with read permission.
tz	Character. Time zone for parsing response timestamps. Default "Asia/Hong_Kong".

Value

A 'data.frame' with data availability and date fields. The nested 'account' and 'programs' API objects are retained as JSON strings.

```
get_var_label
```

Retrieve variable labels from OKX data frames

Description

Returns human-readable labels attached to a data frame produced by OKX API parsers. You can retrieve all labels or a label for a specific variable by name or index.

Usage

```
get_var_label(df, var = NULL, default = NA_character_)
```

Arguments

df	A data.frame with "var_labels" attribute (as returned by OKX API parsers).
var	Optional variable name (character) or index (numeric). If NULL, all labels are returned.
default	Value to return if the variable has no label (default: NA_character_).

Value

A character label if var is provided, or a named character vector of all labels if var = NULL.

Examples

```
df <- data.frame(ordId = "123", px = 10)
attr(df, "var_labels") <- c(ordId = "Order ID", px = "Price")

get_var_label(df, "ordId")
get_var_label(df, 2)
get_var_label(df)
```

post_account_account_level_switch_preset
<i>Preset Account Level Switch</i>

Description

Preset required values before switching account mode.

Usage

```
post_account_account_level_switch_preset(
  acct_lv,
  lever = NULL,
  risk_offset_type = NULL,
  tz = .okx_default_tz,
  config
)
```

Arguments

acct_lv	Target account level.
lever	Optional leverage preset.
risk_offset_type	Optional deprecated risk offset type field.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A ‘data.frame’ describing the stored preset values.

post_account_mmp_config
Configure MMP

Description

Set market maker protection thresholds for an options instrument family.

Usage

```
post_account_mmp_config(  
    inst_family,  
    time_interval,  
    frozen_interval,  
    qty_limit,  
    tz = .okx_default_tz,  
    config  
)
```

Arguments

inst_family	Instrument family.
time_interval	Time window in milliseconds.
frozen_interval	Frozen interval in milliseconds.
qty_limit	Quantity limit in number of contracts.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' describing the applied MMP configuration.

post_account_mmp_reset
Reset MMP Status

Description

Reset market maker protection status for an instrument family.

Usage

```
post_account_mmp_reset(
    inst_family,
    inst_type = NULL,
    tz = .okx_default_tz,
    config
)
```

Arguments

inst_family	Instrument family.
inst_type	Optional instrument type. Defaults to "OPTION" on OKX.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' with the request result.

post_account_move_positions

Move Positions Between Accounts

Description

Move positions between accounts under the same master account.

Usage

```
post_account_move_positions(
    from_acct,
    to_acct,
    legs,
    client_id,
    tz = .okx_default_tz,
    config
)
```

Arguments

from_acct	Source account name.
to_acct	Destination account name.
legs	List of move-position leg objects in the documented OKX shape.
client_id	Client-supplied request ID.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' describing the move-position request result.

post_account_position_margin_balance
<i>Adjust Position Margin Balance</i>

Description

Increase or reduce margin for an existing position.

Usage

```
post_account_position_margin_balance(  
  inst_id,  
  pos_side,  
  type,  
  amt,  
  ccy = NULL,  
  tz = .okx_default_tz,  
  config  
)
```

Arguments

inst_id	Instrument ID.
pos_side	Position side.
type	Margin adjustment type, typically "add" or "reduce".
amt	Amount to add or reduce.
ccy	Optional currency for isolated margin orders.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' describing the applied margin adjustment.

post_account_set_account_level
<i>Set Account Level</i>

Description

Switch the account mode.

Usage

```
post_account_set_account_level(acct_lv, tz = .okx_default_tz, config)
```

Arguments

- | | |
|---------|--|
| acct_lv | Account level string, such as "1", "2", "3", or "4". |
| tz | Timezone used for any timestamp parsing. |
| config | A list containing API credentials. |

Value

A 'data.frame' confirming the applied account level.

post_account_set_auto_loan
<i>Set Account Auto Loan</i>

Description

Enable or disable automatic borrowing.

Usage

```
post_account_set_auto_loan(auto_loan = TRUE, tz = .okx_default_tz, config)
```

Arguments

- | | |
|-----------|---|
| auto_loan | Logical. Whether auto loan should be enabled. |
| tz | Timezone used for any timestamp parsing. |
| config | A list containing API credentials. |

Value

A 'data.frame' confirming the applied auto-loan setting.

post_account_set_auto_repay
Set Account Auto Repay

Description

Enable or disable spot-mode auto repay.

Usage

```
post_account_set_auto_repay(auto_repay, tz = .okx_default_tz, config)
```

Arguments

auto_repay	Logical. Whether auto repay should be enabled.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' confirming the applied auto-repay setting.

post_account_set_collateral_assets
Set Account Collateral Assets

Description

Configure whether all or selected assets are treated as collateral.

Usage

```
post_account_set_collateral_assets(  
  type,  
  collateral_enabled,  
  ccy_list = NULL,  
  tz = .okx_default_tz,  
  config  
)
```

Arguments

type	Type of update, typically "all" or "custom".
collateral_enabled	Logical. Whether the selected assets should be collateral-enabled.
ccy_list	Optional character vector of currencies. Required when 'type = "custom"'.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' confirming the applied collateral asset setting.

post_account_set_fee_type
<i>Set Account Fee Type</i>

Description

Configure the fee charging mode for spot trading.

Usage

```
post_account_set_fee_type(fee_type, tz = .okx_default_tz, config)
```

Arguments

fee_type	Fee type string, typically "0" or "1".
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' confirming the applied fee type.

`post_account_set_greeks`*Set Account Greeks Display Type*

Description

Configure whether Greeks are displayed in PA or BS mode.

Usage

```
post_account_set_greeks(greeks_type, tz = .okx_default_tz, config)
```

Arguments

<code>greeks_type</code>	Greeks display type, typically "PA" or "BS".
<code>tz</code>	Timezone used for any timestamp parsing.
<code>config</code>	A list containing API credentials.

Value

A 'data.frame' confirming the applied Greeks display type.

`post_account_set_leverage`*Set Account Leverage*

Description

Sets the leverage level for a specific trading instrument and margin mode.

Usage

```
post_account_set_leverage(  
  inst_id,  
  lever,  
  mgn_mode,  
  pos_side = NULL,  
  tz = .okx_default_tz,  
  config  
)
```

Arguments

inst_id	Instrument ID (e.g., "BTC-USDT").
lever	Leverage level to apply (as a string or numeric, e.g., "10").
mgn_mode	Margin mode: "cross" or "isolated".
pos_side	Optional. Position side: "long" or "short". Required for isolated mode.
tz	Timezone used for any timestamp parsing (default: "Asia/Hong_Kong").
config	API credential list with keys 'api_key', 'secret_key', and 'passphrase'.

Value

A data.frame with leverage update confirmation (including instrument ID and leverage settings).

```
post_account_set_position_mode
```

Set Account Position Mode

Description

Set the account position mode.

Usage

```
post_account_set_position_mode(pos_mode, tz = .okx_default_tz, config)
```

Arguments

pos_mode	Position mode. Use 'long_short_mode' or 'net_mode'.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' confirming the applied position mode.

post_account_spot_manual_borrow_repay
Submit Spot Manual Borrow or Repay

Description

Manually borrow or repay under spot mode.

Usage

```
post_account_spot_manual_borrow_repay(  
    ccy,  
    side,  
    amt,  
    tz = .okx_default_tz,  
    config  
)
```

Arguments

ccy	Currency.
side	Action side, typically "borrow" or "repay".
amt	Amount.
tz	Timezone used for any timestamp parsing.
config	A list containing API credentials.

Value

A 'data.frame' describing the executed borrow/repay request.

post_asset_cancel_withdrawal
Cancel an Asset Withdrawal

Description

Cancel a pending withdrawal request.

Usage

```
post_asset_cancel_withdrawal(wd_id, tz = .okx_default_tz, config)
```

Arguments

wd_id	Withdrawal request ID.
tz	Timezone used for any timestamp parsing.
config	API credential list.

Value

A 'data.frame' confirming the cancelled withdrawal ID.

post_asset_convert_estimate_quote
<i>Estimate an Asset Convert Quote</i>

Description

Request a quote for an asset conversion without executing the trade.

Usage

```
post_asset_convert_estimate_quote(  
  base_ccy,  
  quote_ccy,  
  side,  
  rfq_sz,  
  rfq_sz_ccy,  
  cl_q_req_id = NULL,  
  tag = NULL,  
  convert_mode = NULL,  
  tz = .okx_default_tz,  
  config  
)
```

Arguments

base_ccy	Base currency.
quote_ccy	Quote currency.
side	Quote side, such as "buy" or "sell".
rfq_sz	RFQ size.
rfq_sz_ccy	Currency in which 'rfq_sz' is specified.
cl_q_req_id	Optional client quote request ID.
tag	Optional request tag.
convert_mode	Optional OKX convert mode.
tz	Timezone used for any timestamp parsing.
config	API credential list.

Value

A 'data.frame' describing the estimated conversion quote.

post_asset_convert_trade

Execute an Asset Convert Trade

Description

Execute a confirmed asset conversion against a previously quoted price.

Usage

```
post_asset_convert_trade(
  quote_id,
  base_ccy,
  quote_ccy,
  side,
  sz,
  sz_ccy,
  cl_t_req_id = NULL,
  tag = NULL,
  convert_mode = NULL,
  tz = .okx_default_tz,
  config
)
```

Arguments

quote_id	Quote ID returned by [post_asset_convert_estimate_quote()].
base_ccy	Base currency.
quote_ccy	Quote currency.
side	Trade side, such as "buy" or "sell".
sz	Trade size.
sz_ccy	Currency in which 'sz' is specified.
cl_t_req_id	Optional client trade request ID.
tag	Optional request tag.
convert_mode	Optional OKX convert mode.
tz	Timezone used for any timestamp parsing.
config	API credential list.

Value

A 'data.frame' describing the executed conversion trade.

post_asset_transfer	<i>Transfer Assets</i>
---------------------	------------------------

Description

Transfer assets between funding, trading, and related accounts.

Usage

```
post_asset_transfer(
  ccy,
  amt,
  from,
  to,
  type = "0",
  sub_acct = NULL,
  loan_trans = NULL,
  omit_pos_risk = NULL,
  client_id = NULL,
  tz = .okx_default_tz,
  config
)
```

Arguments

ccy	Currency to transfer.
amt	Transfer amount.
from	Source account code, such as "6" for funding or "18" for trading.
to	Destination account code.
type	Optional transfer type code. Defaults to "0" for an internal transfer within the same account.
sub_acct	Optional sub-account name when the transfer type requires it.
loan_trans	Optional logical. Whether the transfer should be treated as a loan transfer.
omit_pos_risk	Optional logical. Whether to omit position risk checks where supported by OKX.
client_id	Optional client-supplied transfer request ID.
tz	Timezone used for any timestamp parsing.
config	API credential list.

Value

A 'data.frame' describing the submitted transfer request.

post_asset_withdrawal *Submit an Asset Withdrawal*

Description

Submit a withdrawal request from the OKX funding account.

Usage

```
post_asset_withdrawal(  
    ccy,  
    amt,  
    dest,  
    to_addr,  
    chain = NULL,  
    to_addr_type = NULL,  
    area_code = NULL,  
    rcvr_info = NULL,  
    client_id = NULL,  
    tz = .okx_default_tz,  
    config  
)
```

Arguments

ccy	Currency to withdraw.
amt	Withdrawal amount.
dest	Destination type code from the OKX API.
to_addr	Destination wallet address.
chain	Optional chain identifier, such as <code>"USDT-ERC20"</code> .
to_addr_type	Optional destination address type code.
area_code	Optional phone area code when required by OKX.
rcvr_info	Optional named list in the documented <code>'rcvrInfo'</code> shape.
client_id	Optional client-supplied withdrawal request ID.
tz	Timezone used for any timestamp parsing.
config	API credential list.

Value

A `'data.frame'` describing the submitted withdrawal request.

 post_trade_amend_algos

Amend an Algo Order

Description

Amend a supported unfilled algo order.

Usage

```
post_trade_amend_algos(
    inst_id,
    algo_id = NULL,
    algo_cl_ord_id = NULL,
    cxl_on_fail = NULL,
    req_id = NULL,
    new_sz = NULL,
    new_tp_trigger_px = NULL,
    new_tp_ord_px = NULL,
    new_sl_trigger_px = NULL,
    new_sl_ord_px = NULL,
    new_tp_trigger_px_type = NULL,
    new_sl_trigger_px_type = NULL,
    new_trigger_px = NULL,
    new_ord_px = NULL,
    new_trigger_px_type = NULL,
    attach_algo_ords = NULL,
    adv_chase_params = NULL,
    config,
    tz = .okx_default_tz
)
```

Arguments

inst_id	Instrument ID.
algo_id	Algo order ID. Optional if 'algo_cl_ord_id' is supplied.
algo_cl_ord_id	Client-supplied algo ID. Optional if 'algo_id' is supplied.
cxl_on_fail	Optional logical. Whether to cancel the order if the amendment fails.
req_id	Optional client amendment request ID.
new_sz	Optional new quantity after amendment.
new_tp_trigger_px	Optional new take-profit trigger price.
new_tp_ord_px	Optional new take-profit order price.
new_sl_trigger_px	Optional new stop-loss trigger price.

new_sl_ord_px Optional new stop-loss order price.
 new_tp_trigger_px_type Optional new take-profit trigger price type.
 new_sl_trigger_px_type Optional new stop-loss trigger price type.
 new_trigger_px Optional new trigger price for trigger orders.
 new_ord_px Optional new order price for trigger orders.
 new_trigger_px_type Optional new trigger price type for trigger orders.
 attach_algo_orcs Optional attached TP/SL amendment list.
 adv_chase_params Optional amended chase-order parameters.
 config A list with API credentials.
 tz Timezone for parsing response timestamps.

Value

A 'data.frame' describing the algo amendment result.

post_trade_amend_batch_orders
Amend Multiple Trade Orders

Description

Submit multiple amendment requests in one request.

Usage

```
post_trade_amend_batch_orders(orders, config, tz = .okx_default_tz)
```

Arguments

orders List of amendment specs using the names from [post_trade_amend_order()].
 config A list with API credentials.
 tz Timezone for parsing response timestamps.

Value

A 'data.frame' with one row per amendment result.

 post_trade_amend_order

Amend a Trade Order

Description

Submit an amendment request for an incomplete order.

Usage

```
post_trade_amend_order(
    inst_id,
    ord_id = NULL,
    cl_ord_id = NULL,
    req_id = NULL,
    new_sz = NULL,
    new_px = NULL,
    cxl_on_fail = NULL,
    new_px_usd = NULL,
    new_px_vol = NULL,
    px_amend_type = NULL,
    attach_algo_ords = NULL,
    speed_bump = NULL,
    rpi_taker_access = NULL,
    rpi_px_round = NULL,
    config,
    tz = .okx_default_tz
)
```

Arguments

inst_id	Instrument ID.
ord_id	Order ID. Optional if 'cl_ord_id' is supplied.
cl_ord_id	Client order ID. Optional if 'ord_id' is supplied.
req_id	Optional client amendment request ID.
new_sz	Optional new total order size.
new_px	Optional new price.
cxl_on_fail	Optional logical. Whether to cancel the order if the amendment fails.
new_px_usd	Optional new option order USD price.
new_px_vol	Optional new option order implied volatility price.
px_amend_type	Optional price amendment mode.
attach_algo_ords	Optional attached TP/SL amendment list.
speed_bump	Deprecated. Ignored by OKX and not sent by 'okxr'.

rpi_taker_access	Optional logical. Whether the amended order can take RPI liquidity.
rpi_px_round	Optional logical. Whether an invalid RPI maker price is rounded to the nearest valid non-crossing price.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A 'data.frame' describing the amendment request result.

 post_trade_batch_orders

Place Multiple Trade Orders

Description

Submit multiple trade orders in one request.

Usage

```
post_trade_batch_orders(orders, config, tz = .okx_default_tz)
```

Arguments

orders	List of order specs using the snake_case names from [post_trade_order()].
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A 'data.frame' with one row per submitted order result.

 post_trade_cancel_algos

Cancel Multiple Algo Orders

Description

Cancel up to 10 unfilled algo orders in one request.

Usage

```
post_trade_cancel_algos(orders, config, tz = .okx_default_tz)
```

Arguments

orders	List of cancellation specification lists containing ‘inst_id’ plus either ‘algo_id’ or ‘algo_cl_ord_id’.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A ‘data.frame’ with one row per algo cancellation result.

post_trade_cancel_all_after
<i>Set Cancel-All-After</i>

Description

Set or disable the cancel-all-after countdown.

Usage

```
post_trade_cancel_all_after(time_out, tag = NULL, config, tz = .okx_default_tz)
```

Arguments

time_out	Character or numeric. Countdown in seconds. ‘0’ disables it.
tag	Optional cancel-all-after tag scope.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A ‘data.frame’ with the configured trigger time and tag.

post_trade_cancel_batch_orders
Cancel Multiple Trade Orders

Description

Submit a batch cancellation request for incomplete orders.

Usage

```
post_trade_cancel_batch_orders(orders, config, tz = .okx_default_tz)
```

Arguments

orders	List of cancellation specification lists containing 'inst_id' plus either 'ord_id' or 'cl_ord_id'.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A 'data.frame' with one row per cancellation result.

post_trade_cancel_order
Cancel a Trade Order

Description

Submits a cancellation request for a previously placed trade order.

Usage

```
post_trade_cancel_order(  
  inst_id,  
  ord_id = NULL,  
  cl_ord_id = NULL,  
  config,  
  tz = .okx_default_tz  
)
```

Arguments

inst_id	Instrument ID (e.g., "BTC-USDT").
ord_id	Optional OKX order ID to cancel.
cl_ord_id	Optional client order ID to cancel. Provide this or 'ord_id'.
config	A list with API credentials: api_key, secret_key, passphrase.
tz	Timezone for parsing any timestamps (default: "Asia/Hong_Kong").

Value

A data.frame containing cancellation result and timestamp.

post_trade_close_position
<i>Close a Position</i>

Description

Submits a request to close a position for a given instrument and position side.

Usage

```
post_trade_close_position(
  inst_id,
  mgn_mode,
  pos_side,
  tz = .okx_default_tz,
  config
)
```

Arguments

inst_id	Instrument ID (e.g., "BTC-USDT").
mgn_mode	Margin mode: "cross" or "isolated".
pos_side	Position side to close: "long" or "short".
tz	Timezone for parsing any timestamps (default: "Asia/Hong_Kong").
config	A list with API credentials: api_key, secret_key, passphrase.

Value

A data.frame with close position confirmation details.

`post_trade_mass_cancel`*Mass Cancel MMP Orders*

Description

Cancel all MMP pending orders for an options instrument family.

Usage

```
post_trade_mass_cancel(  
    inst_type,  
    inst_family,  
    lock_interval = NULL,  
    config,  
    tz = .okx_default_tz  
)
```

Arguments

<code>inst_type</code>	Instrument type. Currently 'OPTION'.
<code>inst_family</code>	Instrument family.
<code>lock_interval</code>	Optional lock interval in milliseconds.
<code>config</code>	A list with API credentials.
<code>tz</code>	Timezone for parsing response timestamps.

Value

A 'data.frame' with the request result.

`post_trade_order`*Place a Trade Order*

Description

Submits a trade order to the OKX exchange.

Usage

```

post_trade_order(
    inst_id,
    td_mode,
    side,
    ord_type,
    sz,
    pos_side = NULL,
    px = NULL,
    reduce_only = NULL,
    tgt_ccy = NULL,
    cl_ord_id = NULL,
    tag = NULL,
    is_elp_taker_access = NULL,
    rpi_taker_access = NULL,
    rpi_px_round = NULL,
    slippage_pct = NULL,
    config,
    tz = .okx_default_tz
)

```

Arguments

<code>inst_id</code>	Instrument ID (e.g., "BTC-USDT").
<code>td_mode</code>	Trade mode: "cross" or "isolated".
<code>side</code>	Order side: "buy" or "sell".
<code>ord_type</code>	Order type: "limit", "market", etc.
<code>sz</code>	Size of the order (quantity to buy/sell).
<code>pos_side</code>	Optional. Position side: "long" or "short".
<code>px</code>	Optional. Price (required for limit orders).
<code>reduce_only</code>	Optional. Logical flag to indicate a reduce-only order.
<code>tgt_ccy</code>	Optional. Quote currency (e.g., "base", "quote").
<code>cl_ord_id</code>	Optional. Custom client order ID (auto-generated if NULL).
<code>tag</code>	Optional. Tag used for identifying the strategy or bot.
<code>is_elp_taker_access</code>	Deprecated optional logical. Use 'rpi_taker_access'; supplying both is an error.
<code>rpi_taker_access</code>	Optional logical. Whether the order can take RPI liquidity.
<code>rpi_px_round</code>	Optional logical. Whether an invalid RPI maker price is rounded to the nearest valid non-crossing price.
<code>slippage_pct</code>	Optional maximum acceptable slippage for SPOT and margin market orders.
<code>config</code>	A list with API credentials: <code>api_key</code> , <code>secret_key</code> , <code>passphrase</code> .
<code>tz</code>	Timezone for parsing any timestamps (default: "Asia/Hong_Kong").

Value

A data.frame containing fields like order ID, client order ID, and timestamp.

post_trade_order_algo *Place an Algo Order*

Description

Submit an algo order, including chase trigger orders where supported by OKX.

Usage

```
post_trade_order_algo(
  inst_id,
  td_mode,
  side,
  ord_type,
  sz = NULL,
  algo_cl_ord_id = NULL,
  pos_side = NULL,
  reduce_only = NULL,
  tgt_ccy = NULL,
  tag = NULL,
  trigger_px = NULL,
  order_px = NULL,
  trigger_px_type = NULL,
  advance_ord_type = NULL,
  adv_chase_params = NULL,
  attach_algo_ords = NULL,
  config,
  tz = .okx_default_tz
)
```

Arguments

inst_id	Instrument ID.
td_mode	Trade mode.
side	Order side.
ord_type	Algo order type.
sz	Optional order size.
algo_cl_ord_id	Optional client-supplied algo order ID.
pos_side	Optional position side.
reduce_only	Optional logical reduce-only flag.
tgt_ccy	Optional target currency mode.

tag	Optional order tag.
trigger_px	Optional trigger price for trigger algo orders.
order_px	Optional order price for trigger algo orders.
trigger_px_type	Optional trigger price type.
advance_ord_type	Optional advanced order type, e.g. "chase".
adv_chase_params	Optional list of chase-order parameters required when 'advance_ord_type = "chase"'. Use the OKX field names such as 'chaseType', 'chaseVal', 'maxChaseType', and 'maxChaseVal'; the maximum chase fields must be supplied together.
attach_algo_ords	Optional attached TP/SL list. Not valid with chase algo orders.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A 'data.frame' describing the algo order submission result.

post_trade_order_precheck

Precheck a Trade Order

Description

Submit an order precheck request without placing the order.

Usage

```
post_trade_order_precheck(
  inst_id,
  td_mode,
  side,
  ord_type,
  sz,
  ccy = NULL,
  cl_ord_id = NULL,
  tag = NULL,
  pos_side = NULL,
  px = NULL,
  reduce_only = NULL,
  tgt_ccy = NULL,
  attach_algo_ords = NULL,
```

```

    speed_bump = NULL,
    outcome = NULL,
    is_elp_taker_access = NULL,
    slippage_pct = NULL,
    config,
    tz = .okx_default_tz
)

```

Arguments

inst_id	Instrument ID.
td_mode	Trade mode.
side	Order side.
ord_type	Order type.
sz	Order size.
ccy	Optional margin currency.
cl_ord_id	Optional client order ID.
tag	Optional order tag.
pos_side	Optional position side.
px	Optional order price.
reduce_only	Optional logical reduce-only flag.
tgt_ccy	Optional target currency mode.
attach_algo_orbs	Optional attached TP/SL list.
speed_bump	Deprecated. Ignored by OKX and not sent by 'okxr'.
outcome	Optional event-contract outcome.
is_elp_taker_access	Deprecated optional logical. The RPI access parameter does not apply to this endpoint.
slippage_pct	Optional maximum acceptable slippage for SPOT and margin market orders.
config	A list with API credentials.
tz	Timezone for parsing response timestamps.

Value

A 'data.frame' with projected account metrics after the precheck.

set_okxr_options	<i>Set or get okxr options</i>
------------------	--------------------------------

Description

Convenience wrapper to set global options for okxr, such as whether to return raw data instead of parsed data.

Usage

```
set_okxr_options(raw_data = NULL, timeout = NULL)
```

Arguments

raw_data	Logical. If 'TRUE', return raw API 'data'. If 'NULL', the current value is left unchanged.
timeout	Numeric. HTTP request timeout in seconds. If 'NULL', the current value is left unchanged.

Value

An invisible named list with the current package options: 'raw_data' (logical) and 'timeout' (numeric seconds). This return value can be used to inspect the effective option state after updating it.

Examples

```
old <- getOption("okxr.raw_data")
old_timeout <- getOption("okxr.timeout")
set_okxr_options(raw_data = TRUE)
set_okxr_options(timeout = 5)
options(okxr.raw_data = old, okxr.timeout = old_timeout)

set_okxr_options() # check current values
```

Index

* package

- okxr-package, [5](#)
- .build_request, [6](#)
- .execute_get_action, [7](#)
- .execute_post_action, [7](#)
- .get_headers, [8](#)

- get_account_adjust_leverage_info, [8](#)
- get_account_balance, [9](#)
- get_account_bill_types, [12](#)
- get_account_bills, [10](#)
- get_account_bills_archive, [11](#)
- get_account_bills_history_archive, [12](#)
- get_account_collateral_assets, [13](#)
- get_account_config, [14](#)
- get_account_greeks, [15](#)
- get_account_instruments, [15](#)
- get_account_interest_accrued, [16](#)
- get_account_interest_limits, [17](#)
- get_account_interest_rate, [18](#)
- get_account_leverage_info, [18](#)
- get_account_max_avail_size, [19](#)
- get_account_max_loan, [20](#)
- get_account_max_size, [21](#)
- get_account_max_withdrawal, [22](#)
- get_account_mmp_config, [22](#)
- get_account_move_positions_history, [23](#)
- get_account_position_risk, [26](#)
- get_account_position_tiers, [26](#)
- get_account_positions, [24](#)
- get_account_positions_history, [25](#)
- get_account_precheck_set_delta_neutral, [27](#)
- get_account_risk_state, [28](#)
- get_account_set_account_switch_precheck, [28](#)
- get_account_spot_borrow_repay_history, [29](#)
- get_account_subaccount_balances, [29](#)
- get_account_subaccount_max_withdrawal, [30](#)
- get_account_subtypes, [31](#)
- get_account_trade_fee, [31](#)
- get_asset_asset_valuation, [32](#)
- get_asset_balances, [33](#)
- get_asset_bills, [33](#)
- get_asset_bills_history, [34](#)
- get_asset_convert_currencies, [35](#)
- get_asset_convert_currency_pair, [36](#)
- get_asset_convert_history, [36](#)
- get_asset_currencies, [37](#)
- get_asset_deposit_address, [38](#)
- get_asset_deposit_history, [38](#)
- get_asset_deposit_withdraw_status, [39](#)
- get_asset_exchange_list, [40](#)
- get_asset_non_tradable_assets, [40](#)
- get_asset_transfer_state, [41](#)
- get_asset_withdrawal_history, [41](#)
- get_copy_trade_config, [42](#)
- get_copy_trade_current_subpos, [43](#)
- get_copy_trade_historical_subpos, [44](#)
- get_copy_trade_instruments, [45](#)
- get_copy_trade_my_leaders, [46](#)
- get_copy_trade_profit_sharing_details, [47](#)
- get_copy_trade_public_config, [48](#)
- get_copy_trade_public_copy_traders, [48](#)
- get_copy_trade_public_current_subpositions, [49](#)
- get_copy_trade_public_lead_traders, [50](#)
- get_copy_trade_public_pnl, [51](#)
- get_copy_trade_public_preference_currency, [51](#)
- get_copy_trade_public_stats, [52](#)
- get_copy_trade_public_subpositions_history, [53](#)
- get_copy_trade_public_weekly_pnl, [53](#)
- get_copy_trade_settings, [54](#)

- get_copy_trade_total_profit_sharing, 55
- get_copy_trade_total_unrealized_profit_sharing, 56
- get_copy_trade_unrealized_profit_sharing_details, 56
- get_market_block_ticker, 57
- get_market_block_tickers, 57
- get_market_books, 58
- get_market_books_rpi, 59
- get_market_candles, 59
- get_market_exchange_rate, 60
- get_market_history_candles, 61
- get_market_history_index_candles, 62
- get_market_history_mark_price_candles, 63
- get_market_history_trades, 64
- get_market_index_candles, 65
- get_market_index_components, 66
- get_market_index_tickers, 66
- get_market_mark_price_candles, 67
- get_market_option_instrument_family_trades, 68
- get_market_platform_24_volume, 68
- get_market_ticker, 69
- get_market_tickers, 69
- get_market_trades, 70
- get_public_block_trades, 71
- get_public_convert_contract_coin, 71
- get_public_delivery_exercise_history, 72
- get_public_discount_rate_interest_free_quota, 73
- get_public_economic_calendar, 74
- get_public_estimated_price, 74
- get_public_estimated_settlement_info, 75
- get_public_event_contract_events, 76
- get_public_event_contract_markets, 77
- get_public_event_contract_series, 78
- get_public_funding_rate, 78
- get_public_funding_rate_history, 79
- get_public_instrument_tick_bands, 81
- get_public_instruments, 80
- get_public_insurance_fund, 82
- get_public_interest_rate_loan_quota, 83
- get_public_mark_price, 83
- get_public_mm_instrument_types, 84
- get_public_open_interest, 85
- get_public_opt_summary, 87
- get_public_option_trades, 86
- get_public_position_tiers, 87
- get_public_premium_history, 88
- get_public_price_limit, 89
- get_public_settlement_history, 90
- get_public_time, 90
- get_public_underlying, 91
- get_trade_account_rate_limit, 91
- get_trade_easy_convert_currency_list, 92
- get_trade_easy_convert_history, 93
- get_trade_fills, 93
- get_trade_fills_history, 94
- get_trade_one_click_repay_currency_list, 95
- get_trade_one_click_repay_currency_list_v2, 96
- get_trade_one_click_repay_history, 97
- get_trade_one_click_repay_history_v2, 97
- get_trade_order, 98
- get_trade_order_algo, 105
- get_trade_orders_algo_history, 99
- get_trade_orders_algo_pending, 100
- get_trade_orders_history, 101
- get_trade_orders_history_7d, 102
- get_trade_orders_history_archive, 103
- get_trade_orders_pending, 104
- get_user_glp_historical_performance, 106
- get_user_glp_today_performance, 107
- get_var_label, 107
- okxr (okxr-package), 5
- okxr-package, 5
- post_account_account_level_switch_preset, 108
- post_account_mmp_config, 109
- post_account_mmp_reset, 109
- post_account_move_positions, 110
- post_account_position_margin_balance, 111
- post_account_set_account_level, 112
- post_account_set_auto_loan, 112
- post_account_set_auto_repay, 113

post_account_set_collateral_assets,
 [113](#)
post_account_set_fee_type, [114](#)
post_account_set_greeks, [115](#)
post_account_set_leverage, [115](#)
post_account_set_position_mode, [116](#)
post_account_spot_manual_borrow_repay,
 [117](#)
post_asset_cancel_withdrawal, [117](#)
post_asset_convert_estimate_quote, [118](#)
post_asset_convert_trade, [119](#)
post_asset_transfer, [120](#)
post_asset_withdrawal, [121](#)
post_trade_amend_algos, [122](#)
post_trade_amend_batch_orders, [123](#)
post_trade_amend_order, [124](#)
post_trade_batch_orders, [125](#)
post_trade_cancel_algos, [125](#)
post_trade_cancel_all_after, [126](#)
post_trade_cancel_batch_orders, [127](#)
post_trade_cancel_order, [127](#)
post_trade_close_position, [128](#)
post_trade_mass_cancel, [129](#)
post_trade_order, [129](#)
post_trade_order_algo, [131](#)
post_trade_order_precheck, [132](#)

set_okxr_options, [134](#)