Security Model

The security model contains descriptive information about all securities in our holdings. A security refers to any of the following:

1) Equity;

2) Bond;

3) Fund;

4) Futures Contract;

5) Fixed Deposit;

6) FX Forward;

7) Repo.

Here we divide the descriptive information into 2 parts, i.e., security basics and security attributes.

# Security Basics

Here we have a few data structures that describe the basic information of a security in Geneva system.

## Security Id and Type

This data structure describes the security id, type, name and related stuff.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| Geneva Id | String | Geneva investment id. This is a unique id. | “700 HK” |
| Geneva Asset Type | String | Geneva asset type | “Equities” |
| Geneva Investment Type | String | Geneva investment type | “Common Stock” |
| Ticker | String | Bloomberg Ticker | “700 HK Equity” |
| ISIN | String | ISIN Code | “KYG875721634” |
| Bloomberg Id | String | Bloomberg ID | “BBG000BJ35N5” |
| SEDOL | String | SEDOL Code | “BMMV2K8” |
| Currency | String | Currency of the investment | “HKD” |
| Is Private | String | Indicate whether this is a private equity, or fund. “Y” means yes, “N” means no, “NA” means information not available. | “N”, “Y”, “NA” |
| Description | String | Name of the security, fund, or futures contract | “Tencent Holdings” |
| Exchange Name | String | Exchange name | “HKEX” |
| Time Stamp | String | The time when this record is inserted or updated | “2021-03-31 15:20:10” |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_security\_basic\_info | [String] Geneva investment id | [Dictionary] security info |
| add\_security\_basic\_info | [Dictionary] security info | Raise error “security info exists”, if the Geneva id already exists in the underlying data store. |
| update\_security\_basic\_info | [Dictionary] security info | Raise error “security info not found”, if the Geneva id does not exist in the underlying data store. |

NOTE:

1) When a new record is added, or an existing one updated, the three fields: “Geneva Id”, “Geneva Asset Type”, and “Geneva Investment Type” cannot be empty. If they are, raise error “invalid security info”.

2) The timestamp field is not included in the info object passed to add\_security\_basic\_info(), or update\_security\_basic\_info() function calls. Rather, it is the time when the underlying datastore is updated.

3) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. For example, passing the below dictionary to update\_security\_basic\_info() is valid, which means only update the “ISIN” and “Is Private” fields for the record with “Geneva Id” = “700 HK”.

|  |  |
| --- | --- |
| Key | Value |
| Geneva Id | “700 HK” |
| ISIN | “XS1234567890” |
| Is Private | “N” |

## Futures

This data structure contains some more information on futures contracts.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| Ticker | String | Bloomberg Ticker. This is a unique id. | “TYM1 Comdty” |
| Underlying Id | String | Underlying Security Id | “US 10yr 6%” |
| Contract Size | Float | Contract Size | 100,000 |
| Value of 1pt | Float | Value change when the price changes 1.0 | 1,000 |
| Time Stamp | String | The time when this record is inserted or updated | “2021-03-31 15:20:10” |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_futures\_info | [String] Ticker | [Dictionary] security info |
| add\_ futures\_info \_info | [Dictionary] security info | Raise error “security info exists”, if the Ticker already exists in the underlying data store. |
| update\_ futures\_info \_info | [Dictionary] security info | Raise error “security info not found”, if the Ticker does not exist in the underlying data store. |

NOTE:

1) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().

## Fixed Deposit

This data structure contains information on fixed deposits.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| Geneva Id | String | Geneva investment id. This is a unique id | “IB Fixed Deposit 0.651 07/08/2021” |
| FactSet Id | String | FactSet security id | “IB\_Fixed\_Deposit\_0\_pt\_651\_07082021” |
| Geneva Counter Party | String | The bank that keeps the fixed deposit | “IB” |
| Starting Date | String | The starting date of fixed deposit | “2021-01-08” |
| Maturity Date | String | The maturity date of fixed deposit | “2021-07-08” |
| Interest Rate | Float | Interest rate | 0.75 |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_fixed\_deposit\_info | [String] Geneva Id | [Dictionary] security info |
| add\_fixed\_deposit\_info | [Dictionary] security info | Raise error “security info exists”, if the Ticker already exists in the underlying data store. |
| update\_fixed\_deposit\_info | [Dictionary] security info | Raise error “security info not found”, if the Ticker does not exist in the underlying data store. |

NOTE:

1) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().

2) When add\_fixed\_deposit\_info() is called, it not only adds a record to the underlying datastore, but also calls the add\_counter\_party() function (see “OTC Counter Party” section), with “Geneva Counter Party” = Geneva Counter Party, “Counter Party Type” = “Fixed Deposit”. If that function raises a “counter party exists” error, ignore it. Otherwise bubble up the error.

3) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().

## FX Forward

This data structure contains information on FX Forward contacts.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| FactSet Id | String | FactSet security id for FX Forward. This is a unique id. | “FXForward\_1163847” |
| Geneva FX Forward Name | String | Geneva FX Forward Name | "CNH per USD @ 6.55 NOMURA - 07/28/2021 40017" |
| Geneva Counter Party | String | The counter party of the FX Forward contract | “INST-FI” |
| Starting Date | String | When the FX Forward contract was booked | “2021-04-21” |
| Maturity Date | String | When the fixed deposit matures | “2021-05-17” |
| Base Currency | String | The currency to be converted | “USD” |
| Base Currency Quantity | Float | Notional amount in terms of base currency | 348241.37 |
| Term Currency | String | The currency to convert to | “CNH” |
| Term Currency Quantity | Float | Notional amount in terms of term currency | 2350000.00 |
| Forward Rate | Float | The exchange rate | 6.6051 |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_fx\_forward\_info | [String] FactSet Id | [Dictionary] security info |
| add\_fx\_forward\_info | [Dictionary] security info | Raise error “security info exists”, if the FactSet Id already exists in the underlying data store. |
| update\_fx\_forward\_info | [Dictionary] security info | Raise error “security info not found”, if the FactSet Id does not exist in the underlying data store. |

NOTE:

1) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().

2) When add\_fx\_forward\_info() is called, it not only adds a record to the underlying datastore, but also calls the add\_counter\_party() function (see “OTC Counter Party” section), with “Geneva Counter Party” = Geneva Counter Party, “Counter Party Type” = “FX Forward”. If that function raises a “counter party exists” error, ignore it. Otherwise bubble up the error.

3) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().

## OTC Counter Party

This data structure contains information on all OTC counter parties.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| Geneva Counter Party | String | The counter party name | “IB”, “INST-FI”, “BNP-REPO” |
| Counter Party Type | String | Only “FX Forward”, “Fixed Deposit”, and “Repo” are allowed.  Combining “Geneva Counter Party” and “Counter Party Type” forms a unique id. | “Fixed Deposit”, “Repo”, “FX Forward” |
| Counter Party Name | String | Name of the counter party | “Industrial Bank of China” |
| Bloomberg Ticker | String | Bloomberg Ticker for the entity | “601166 CH” |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_all\_counter\_party | No argument | [List] ([Dictionary] counter party info) |
| add\_counter\_party\_info | [Dictionary] counter party info | Raise error “counter party exists”, if the joint key “Geneva Counter Party” and “Counter Party Type” already exists in the underlying datastore. |
| update\_counter\_party\_info | [Dictionary] counter party info | Raise error “counter party not found”, if the joint key does not exist in the underlying datastore. |

NOTE:

1) When a new record is added, or an existing one updated, the two fields: “Geneva Counter Party” and “Counter Party Type” cannot be empty. If they are, raise error “invalid counter party info”.

# Security Attributes

For equity and bond type of securities, we need the below data structure to store more attributes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field** | **Type** | **Meaning** | **Sample** |
| Security Id Type | String | Type of security id. Only “Ticker”, “ISIN”, and “Bloomberg Id” are allowed. | “ISIN” |
| Security Id | String | Security id  Combining “Security Id Type” and “Security Id” forms a unique id. | “XS1936784161” |
| GICS Sector | String | Industry sector based on GICS standard | “Financials” |
| GICS Industry Group | String | Industry group based on GICS standard | “Banks” |
| Industry Sector | String | Industry sector Bloomberg | “Financial” |
| Industry Group | String | Industry group Bloomberg | “Banks” |
| BICS Sector Level 1 | String | Industry sector based on BICS standard | “Financial” |
| BICS Industry Group Level 2 | String | Industry group based on BICS standard, level 2 | “Banks” |
| BICS Industry Name Level 3 | String | Industry group based on BICS standard, level 3 | “” |
| BICS Sub Industry Name Level 4 | String | Industry group based on BICS standard, level 4 | “” |
| Parent Symbol | String | The issuer symbol | “CEHIOZ CH” |
| Parent Symbol Chinese Name | String | The issuer Chinese name | “中央匯金投資  有限責任公司” |
| Parent Symbol Industry Group | String | Industry group of the issuer | “Investment Companies” |
| Cast Parent Company Name | String | The name of the immediate issuer | “China Construction Bank Corp” |
| Country of Risk | String |  | “CN” |
| Country of Issuance | String |  | “CN” |
| SFC Region | String | From API: get\_sfc\_region() | “China Mainland” |
| S&P issuer Rating | String | Credit rating of the issuer S&P | “A” |
| Moody's Issuer Rating | String | Credit rating of the issuer Moody’s | ““ |
| Fitch's Issuer Rating | String | Credit rating of the issuer Fitch | “A” |
| Bond or Equity Ticker | String | Bloomberg Ticker | “CCB” |
| S&P Rating | String | Credit rating from S&P | “BBB+” |
| Moody’s Rating | String | Credit rating from Moody’s | “” |
| Fitch Rating | String | Credit rating from Fitch | “BBB+” |
| Payment Rank | String | Bond payment rank | “Subordinated” |
| Payment Rank MBS | String | Bond payment rank (MBS only) | “” |
| Bond Classification | String | To identify bond belong to Chengtou (城投债) | “” |
| Local Government (LGFV) | String | If it is Chengtou bond, then which local government it belongs to | “Beijing” |
| First Year Default Probability | Float | The probability that the bond will default in the subsequent year | 0.000172683 |
| Contingent Capital | String | For convertible Bond only |  |
| CoCo Bond Trigger | String | For convertible Bond only |  |
| Capital Type Contingent Conversion Trigger level | String | For convertible Bond only |  |
| Tier 1 Common Equity Ratio | Float | For convertible Bond only |  |
| Bail-in Capital Indicator | String | For convertible Bond only |  |
| TLAC / MREL Designation | String | For convertible Bond only |  |
| Classification on Chinese State Owned Enterprise | String | If the bond is issued by Chinese statement owned enterprise, then return the entity type | “Sovereign” |
| Private Placement Indicator | String | For Bond only  Whether the bond is a private placement, “Y” means yes, “N” means no | “N” |
| Trading volume 90 Days | Float | For Equity only  Past 90 days average trading volume | 24,634,300.00 |

## API

Here are functions to get, add and update the security id and type data structure.

|  |  |  |
| --- | --- | --- |
| **Function** | **Arguments** | **Results** |
| get\_security\_attribute | [String] security id type,  [String] security id | [Dictionary] security attribute |
| add\_security\_attribute | [Dictionary] security attribute | Raise error “security attribute exists”, if the joint key “Security Id Type” and “Security Id” already exists in the underlying datastore. |
| update\_security\_attribute | [Dictionary] security attribute | Raise error “security attribute not found”, if the joint key does not exist in the underlying datastore. |

NOTE:

1) When a new record is added, or an existing one updated, the two fields: “Security Id Type” and “Security Id” cannot be empty. If they are, raise error “invalid security attribute”.

2) When a record is updated, keys in the info object not necessarily consist of all the keys in the data structure. Follow a similar logic as update\_security\_basic\_info().