

Institutional Application Program Interface (API)

The URL generally has the form `/api/<resource>/<action>`, which could either retrieve a result or perform some action.

When retrieving a result, the API expects a HTTP GET method, and when performing an action it expects a HTTP POST.

The resource `/api/institutional` is an API specifically designed for institutional clients.

Successful results are returned as HTTP 200 or 201 responses, with the body containing JSON data. Defined error responses will be in the HTTP 400 range, and undefined server errors in the HTTP 500 range.

Error responses contain the following JSON format:

```
{
  "errors" : [
    {
      "description" : "<error description>",
      "code" : "<error code>",
      "index" : <integer>
    }
  ]
}
```

where the error description is a detailed description of the error, and the error code is a short reference to the error.

Index is an optional error field and is present when the request data is an array of objects. It indicates the position of the offending object in the request array.

All date parameters must be in ISO format: YYYY-MM-DD

Supported API calls

- 1 Products**
- 2 Contracts**
- 3 Holdings**
- 4 Get instrument instructions**
- 5 Get instrument instruction details**
- 6 Update entity compliance requirement**

1 Products

GET: /api/institutional/products

Example JSON response:

```
"data": [{"Unique id": 609157888,
  "Name": "Discretionary"},
  {"Unique id": 896356096,
  "Name": "Retirement annuity"}]
```

2 Contracts

This call retrieves all client contracts for the specified product.

GET: /api/institutional/contracts?product={productUniqueId}

Example JSON response:

```
"data" : [{"Unique Id" : 123456,
  "Contract number" : "C23456",
  "Owner unique id" : 3456789,
  "Owner" : "John Doe",
  "Owner identification number" : "6611235090089",
  "Preferred currency" : "ZAR"}]
```

Errors:

HTTP response	Code	Description
400	NotFound	Product not found

3 Holdings

Get holdings of all client accounts in the specified product for a specific date.

GET: /api/institutional/holdings?product={productUniqueId}&date={date}

Example JSON response:

```
"data" : [{"Contract id" : 123456,
  "Client account id" : 234567,
  "Instrument id" : 345678,
  "Units" : {"Instrument id" : 345678,
    "type" : "Unit",
    "value" : "100.0000"},
  "Latest available price" : {"Instrument id" : 345678,
    "currency" : "ZAR",
    "type" : "Price",
    "value" : "12.3456"},
  "Price date" : "2024-06-01",
  "Market Value in Fund Currency" : {"currency" : "ZAR",
    "type" : "Money",
    "value" : "1234.56"},
  "Market Value in System Currency" : {"currency" : "ZAR",
    "type" : "Money",
    "value" : "1234.56"},
  "Contract number" : "C000123",
  "Instrument name" : "Allan Gray Balanced Fund",
  "Instrument code" : "AGBF",
  "Unrealised gains" : {"currency" : "ZAR",
    "type" : "Money",
    "value" : "123.45"}}]
```

Errors:

HTTP response	Code	Description
400	MalformedRequest	Invalid date
404	NotFound	Could not find product

4 Get instrument instructions

Get a list of instrument instructions created in the specified date range.

GET: /api/institutional/instrumentInstructions?
from_date={fromDate}&to_date={toDate}

Example JSON response:

```
"data" : [{"Unique id" : 123456,  
            "Status" : "Created",  
            "Type" : "Buy"}]
```

Errors:

HTTP response	Code	Description
400	MalformedRequest	Invalid date

5 Get instrument instruction details

Get details of an instrument instruction.

GET: `/api/institutional/instrumentInstructions/{uniqueId}`

Example JSON response:

```
{
  "Unique id" : 123456,
  "Status" : "Created",
  "Type" : "Buy",
  "Instrument code" : "AGBF",
  "Instrument unique id" : 234567,
  "Instrument name" : "Allan Gray Balanced Fund",
  "Trading currency" : "ZAR",
  "Transaction date" : "2024-06-02",
  "Date confirmed" : "2024-06-03T14:15:16+02:00",
  "Date created" : "2024-06-02T14:15:16+02:00",
  "Price" : {"Instrument id" : 234567,
    "currency" : "ZAR",
    "type" : "Price",
    "value" : "12.3456"},
  "Units" : {"Instrument id" : 234567,
    "type" : "Unit",
    "value" : "100.0000"},
  "Amount" : {"currency" : "ZAR",
    "type" : "Money",
    "value" : "1234.56"},
  "Allocations" : [{
    "Contract unique id" : 345678,
    "From units" : {"currency" : "ZAR",
      "type" : "Money",
      "value" : "1234.56"},
    "To units" : {"Instrument id" : 234567,
      "type" : "Unit",
      "value" : "100.0000"},
    "Instruction type" : "Switch",
    "Instruction purpose" : "Bulk Rebalance"
  ]
}
```

Errors:

HTTP response	Code	Description
404	NotFound	Could not find instrument instruction

6 Update entity compliance requirement

Set a specific compliance requirement for an entity to be compliant or not.

POST: /api/institutional/entityComplianceRequirements/{entityUniqueId}

Example JSON request:

```
{
  "Requirement name" : "Overall Compliant",
  "Is compliant" : false,
  "Outstanding" : true,
  "Reason" : "ID unclear"
}
```

A response of 200 means that the entity's compliance requirement has been updated.

Errors:

HTTP response	Code	Description
404	NotFound	Could not find entity
404	ComplianceRequirementNotFound	Could not find requirement
404	OutstandingReasonNotFound	Reason not defined on the requirement type

7 Create Switch

Create a switch instruction

Refer to: [Instructions Application Program Interface \(API\) Master](#) : Create Switch

as this api is a pass through to that api

POST: /api/institutional/switches/{contractNumber}

Example JSON request:

Example request to switch using amount for sell:

```
{
  "purpose" : "optional instruction purpose"
  "sellUsing" : "amount"
  sell : [
    {
      "instrument" : "AGNBFB" ,
      "amount" : 500 ,
    },
  ],
}
```

```
{
  "instrument" : "AGNFSA",
  "amount" : 200.01 ,
}
]
buy : [
  {
    "instrument" : "AGNBFD",
    "percentage" : 40 ,
  },
  {
    "instrument" : "AGNSFD",
    "percentage" : 60 ,
  },
]
}
```

JSON success response

HTTP 200

```
{
  "instructionId" : 27647488,
  "instructionUrl" : "URL to the fetch the instruction"
}
```

A response of 200 means that the switch has been created.

Errors:

Refer to: [Instructions Application Program Interface \(API\) Master](#) : Create Switch