Peter Gustafson
writing sample
Upvest Jun 4, 2021

Placing an order · Upvest API Documentation
This guide walks you through all aspects of placing an order with the
Investment API, including the related cash balance and position
updates.

Prerequisites
─────────────

You can always place an order with our Investment API regardless of
whether:

∗   The user checks are passed (know-your-customer, proof-of-
residence, instrument fit).
∗   The account is currently locked for any reason (e.g., compliance
reasons).
∗   The user has sufficient funds in their account.
∗   The exchange is open.
∗   The ISIN is open for trading that day.

However, the order will only be executed when all the previous
requisites are met.

INFO

Conducting an instrument fit check means that to offer investment
products to your users, you are required to assess their investment
experience and knowledge for different instruments.

How to place an order
─────────────────────

We will explain two use cases to show you how the Investment API works
and what you can expect when placing an order:

1. [Placing a nominal buy order](#1-Implementing-a-nominal-buy-order)
(EUR 1,000).
2. [Placing a unit sell order](#2-Implementing-a-unit-sell-order) (10
shares).

INFO

Orders can be created as nominal orders, i.e. buying/selling a
security for a specified amount (e.g. EUR 1,000), or as unit orders,
i.e. buying/selling a security by specifying the quantity of

securities (e.g. 10 shares).

Additional cash / security buffers

For nominal sell and unit buy orders, there is an additional +10% buffer:

*   **For nominal sell orders:** To account for possible price fluctuations, Upvest adds a 10% buffer, i.e. the order is processed if it is estimated to be no more than 90% of the user's current position.
*   **For unit buy orders:** Similar to above, Upvest adds a 10% buffer to ensure sufficient cash balance to purchase securities, i.e. the order will be processed if it is estimated to be no more than 90% of the user's cash balance.

INFO

Orders may be cancelled by Upvest in the event of insufficient cash balance or insufficient securities. However, it is important to note that if an order is cancelled by Upvest, the order can never be executed or placed again, so you can safely place the order again.

1\. Implementing a nominal buy order

How nominal orders work

The following flow diagram illustrates the nominal buy order flow with two equal order executions (execution price of EUR 85.22), including all related state changes, cash balance and positions updates, as well as buy order report webhooks.

![Nominal buy order flowchart](https://doctave-assets-prod.s3.eu-central-1.amazonaws.com/assets/2/21/QSYi6OEyrJhDzvKytPbvAZKw-obiIA9Xx6UCQ67pw8Q/nominal_buy_with_execution.png?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIA5SVLUIAJSXFDENFN%2F20240129%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Date=20240129T000000Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=dfb093a56375e3b58f33d30bf221deddbf588b81d6c3910ad9f6939b5b1de0fb)

INFO

For nominal sell and unit buy orders, there will be one additional webhook:

*   **Nominal sell**: Positions update webhook to unlock the 10% buffer
*   **Unit buy**: Cash balance update webhook to unlock the 10% buffer

Placing a nominal buy order

To place a nominal buy order, send

POST [/orders](https://docs.upvest.co/api/Orders#place-an-order)

Example request

```
{
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
  "cash_amount": "1000",
  "currency": "EUR",
  "side": "BUY",
  "instrument_id": "US0378331005",
  "instrument_id_type": "ISIN",
  "order_type": "MARKET",
  "user_instrument_fit_acknowledgement": true,
  "limit_price": "",
  "stop_price": ""
}
```


For the nominal buy order call, you need to provide the following attributes in the request payload. Any other required order data is populated by Upvest.

* Parameter: cash_amount
  * Description: Required: Specify the cash amount. You can omit the share quantity (vice versa for a unit order), "BUY"` as order side
* Parameter: currency
  * Description: Required: Since each account group will have the cash balance for a specific currency, the currency for the order should be supported by that account group.
* Parameter: instrument_id
  * Description: Required: Specify the specific ISIN you want to trade in.
* Parameter: instrument_id_type

* Description: Required: Specify ISIN as instrument ID type. NOTE: Soon we will also offer UUIDs so that the instrument id type gives you the freedom to interact with our API using UUIDs or ISINs depending on your needs and whatever is easiest for you.
* Parameter: client_reference
  * Description: Optional: you can also provide your own generated order ID. This, for example, allows you to map orders back to your end-users in an easy manner. NOTE: It does not have to be your order ID, it can be anything that you deem relevant, i.e. there is no guarantee that this value is unique.
* Parameter: user_instrument_fit_acknowledgement
  * Description: Optional: Must be set to true if the user has proven to you a non-instrument fit, but has accepted the related risks of which you have informed them. In all other cases it should be false. NOTE: The absence of it will be equal to false. Consequently, Upvest is only interested in the outcome and needs to be able to prove whether an instrument fit check has been performed or not.
* Parameter: order_type
  * Description: Required: The Investment API is designed to support MARKET, LIMIT, and STOP (Market) orders. NOTE: LIMIT and STOP orders can only be placed as full unit orders, i.e. you can only Buy/Sell 1, 2, 3 shares of an instrument.
* Parameter: limit_price
  * Description: Conditional: Must be specified for LIMIT orders. The order is to be executed at the specified price (or better).
* Parameter: stop_price
  * Description: Conditional: Must be specified for STOP orders. The trigger price at which the order is automatically placed as a market order. . NOTE: Placing a Stop Buy order with a stop_price below the current market price will result in an order cancellation because the market price might be more than the available cash amount on the account.
* Parameter: expiry_date
  * Description: Optional: For LIMIT and STOP orders it is the validity of order as the last day the order can trade (default is good-until-cancelled in case of absence of this field).

After a successful order request, the sample response for the above will look as follows:

Example response

```
{
  "id": "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
  "created_at": "2021-07-21T14:10:00.00Z",
  "updated_at": "2021-07-21T14:10:00.00Z",
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
```

```
  "cash_amount": "1000",
  "currency": "EUR",
  "side": "BUY",
  "instrument_id": "US0378331005",
  "instrument_id_type": "ISIN",
  "order_type": "MARKET",
  "quantity": "0",
  "user_instrument_fit_acknowledgement": true,
  "limit_price": "",
  "stop_price": "",
  "status": "NEW",
  "fee": "0.0",
  "executions": [],
  "client_reference": "",
  "initiation_flow": "API"
}
```

Once all relevant order checks have been passed, the order transitions from `NEW` to the `PROCESSING` status and you will receive a webhook for an order event. At this point, the order can no longer be cancelled via the Investment API.

In addition to the `status` field, you will also see the `initiation_flow` used during order creation. The different values will give you insight into what triggered the order:

INFO

The values you can see depend on whether a certain feature is activated for you (i.e Portfolio, Fee, and Reinvestment functionality).

* Parameter: API
  * Description: The order was created via the POST /orders endpoint.
* Parameter: PORTFOLIO
  * Description: An order is a result of a portfolio order placed via the POST /portfolios/orders endpoint.
* Parameter: PORTFOLIO_REBALANCING
  * Description: An order as a result of rebalancing to achieve the target weight of a given portfolio.
* Parameter: SELL_TO_COVER_FEES
  * Description: A sell order as a result of selling a position to collect fees from end users.

* Parameter: SELL_TO_COVER_TAXES
  * Description: A sell order as a result of selling a position to take in pre-payment income taxes, i.e. 'Vorabpauschale', from end users (only applicable to ETFs and mutual funds).
* Parameter: CASH_DIVIDEND_REINVESTMENT
  * Description: An order as a result of automatically reinvested cash dividends.

After the checks the order gets executed at our trading partner and you will receive an order execution `FILLED` webhook for each execution, followed by an order `FILLED` webhook once the order is completely filled, including the amount of fees charged (if applicable). This means that `executions` will contain all order executions associated with this order.

Additionally, taxes get incurred with `"type": "TOTAL"` for buy orders where financial transaction taxes apply. This means that `cash_amount + taxes.amount` is deducted from the cash balance. You will also be able to see the total amount paid for such orders in the cash balance update event. For taxes incurred on sell orders see [Placing a unit sell order](https://docs.upvest.co/guides/orders/placing_an_order#2-implementing-a-unit-sell-order).

2\. Implementing a unit sell order

How unit sell orders work

The following flow diagram illustrates the unit sell order flow with two equal order executions (at an execution price of EUR 85.22), including all related state changes, cash balance and positions updates, as well as sell order report webhooks.

![Unit sell](https://doctave-assets-prod.s3.eu-central-1.amazonaws.com/assets/2/21/4yLxRaVNHwSVffNTUz-I0qATvH2F9eklSrp6LdZt_Lk/unit_sell_with_execution.png?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIA5SVLUIAJSXFDENFN%2F20240129%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Date=20240129T000000Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=8a54a82fef4a2c1d9b242ab3487835e5eb2efc542db348cec1a8872bba7b5326)

Placing a unit sell order

To place a unit sell order send

POST [orders](https://docs.upvest.co/api/Orders#place-an-order)

Example request

```
{
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
  "currency": "EUR",
  "side": "SELL",
  "instrument_id": "US0378331005",
  "instrument_id_type": "ISIN",
  "order_type": "MARKET",
  "quantity": "10",
  "user_instrument_fit_acknowledgement": true,
  "limit_price": "",
  "stop_price": ""
}

```

Unlike the nominal buy order explained above, you must now specify the share quantity (and omitting the cash amount) as well as the `SELL` side.

After a successful order request, the sample response for the above will look as follows:

Example response

```
{
  "id": "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
  "created_at": "2021-07-21T14:10:00.00Z",
  "updated_at": "2021-07-21T14:10:00.00Z",
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
  "cash_amount": "0",
  "currency": "EUR",
  "side": "SELL",
  "instrument_id": "US0378331005",
  "instrument_id_type": "ISIN",
  "order_type": "MARKET",
  "quantity": "10",
  "user_instrument_fit_acknowledgement": true,
  "limit_price": "",
  "stop_price": "",
  "status": "NEW",
  "fee": "0.0",
  "executions": [],
  "client_reference": "",
```

```
  "initiation_flow": "API"
}
```


Analogous to the nominal buy order, you will receive an order
`PROCESSING` webhook, followed by an order execution `FILLED` webhook
for each execution, and an order `FILLED` webhook once the order is
completely filled, including the amount of fees charged (if
applicable), and the taxes incurred:

```
{
  "id": "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
  "created_at": "2021-07-21T14:10:00.00Z",
  "updated_at": "2021-07-21T14:10:00.00Z",
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
  "cash_amount": "0",
  "currency": "EUR",
  "side": "SELL",
  "instrument_id": "US0378331005",
  "instrument_id_type": "ISIN",
  "order_type": "MARKET",
  "quantity": "10",
  "user_instrument_fit_acknowledgement": true,
  "limit_price": "",
  "stop_price": "",
  "status": "FILLED",
  "fee": "0.0",
  "executions": [
    {
      "id": "b9dc0676-8a7d-412d-802a-3b325eefd15e",
      "side": "SELL",
      "currency": "EUR",
      "status": "FILLED",
      "order_id":  "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
      "cash_amount":"852.20",
      "share_quantity":"10",
      "price":"85.22",
      "transaction_time":"2021-07-21T14:10:00.020Z",
      "taxes": [
        {
          "type": "TOTAL",
          "amount": "23.45"
        }
      ],
      "venue_id": "20d6024b-2df4-41ae-8d42-62e4744e455b"
    }
```

```
    ],
    "client_reference": "",
    "initiation_flow": "API"
}
```


INFO

The cash amount is the total amount the user realised from the sell
order, including all taxes incurred.

Similar to the nominal buy order, you will also receive a positions
and cash balance updates respectively, which you can also find in the
API References for [cash balance events](https://docs.upvest.co/api/
Cash-Balances#cash-balance-events) and [positions events](https://
docs.upvest.co/api/Positions#positions-events).

3\. Listing an order by ID

You can retrieve an individual order by sending

GET [/orders/{order_id}](https://docs.upvest.co/api/Orders#get-
an-order-by-id)

4\. Listing order execution by ID

You can retrieve individual order executions via

GET [/orders/{order_id}/executions/{execution_id}](https://
docs.upvest.co/api/Orders#get-an-order-execution-by-id)

Hence, the executions array for the above order looks as follows:

Example response

```
{
  "id": "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
  "created_at": "2021-07-21T14:10:00.00Z",
  "updated_at": "2021-07-21T14:10:00.00Z",
  "user_id": "2dedfeb0-58cd-44f2-ae08-0e41fe0413d9",
  "account_id": "debf2026-f2da-4ff0-bb84-92e45babb1e3",
  "cash_amount": "1000",
  "currency": "EUR",
  "side": "BUY",
  "instrument_id": "US0378331005",
```

```
    "instrument_id_type": "ISIN",
    "order_type": "MARKET",
    "quantity": "0",
    "user_instrument_fit_acknowledgement": true,
    "limit_price": "",
    "stop_price": "",
    "status": "FILLED",
    "fee": "0.0",
    "executions": [
      {
        "id": "b9dc0676-8a7d-412d-802a-3b325eefd15e",
        "side": "BUY",
        "currency": "EUR",
        "status": "FILLED",
        "order_id":  "eb5ba93f-5dfe-4bf1-8571-4da0caacc80c",
        "cash_amount": "1000",
        "share_quantity": "11.734334663",
        "price": "85.22",
        "transaction_time": "2021-07-21T14:10:00.020Z",
        "taxes": [
          {
            "type": "TOTAL",
            "amount": "0"
          }
        ],
        "venue_id": "20d6024b-2df4-41ae-8d42-62e4744e455b"
      }
    ],
    "client_reference": "",
    "initiation_flow": "API"
}

```

Once the order executions are settled (typically on T+2) you will also receive an order execution `SETTLED` webhook. For sell orders, this means that the cash can now be withdrawn (see example 2).

To see what the respective webhooks for cash balance and positions updates look like, please go to the API References for [cash balance events](https://docs.upvest.co/api/Cash-Balances#cash-balance-events) and [positions events](https://docs.upvest.co/api/Positions#positions-events).

Next steps

In order to provide your users with all relevant reports that will be generated when successfully placing an order, you can now move on to

[implement user reporting](https://docs.upvest.co/guides/reports/index_user_reports).