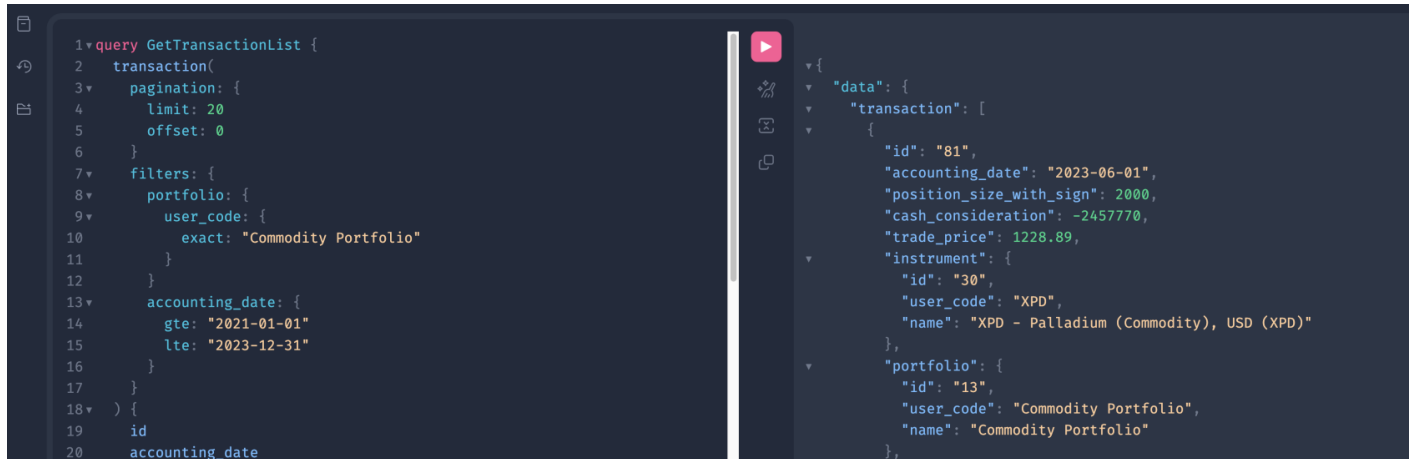


Get Transactions in GraphQL

This query returns transactions for one portfolio and an accounting date range.



The screenshot shows a GraphQL IDE with a query on the left and its JSON response on the right. The query is a GraphQL query named 'GetTransactionList' that filters transactions by portfolio ('Commodity Portfolio') and accounting date range ('2021-01-01' to '2023-12-31'). The response is a JSON object with a 'data' field containing a list of transactions. The first transaction shown has an ID of '81', an accounting date of '2023-06-01', a position size of 2000, a cash consideration of -2457770, and a trade price of 1228.89. The instrument is identified as 'XPD - Palladium (Commodity), USD (XPD)' and the portfolio is 'Commodity Portfolio'.

```
1 query GetTransactionList {
2   transaction(
3     pagination: {
4       limit: 20
5       offset: 0
6     }
7     filters: {
8       portfolio: {
9         user_code: {
10          exact: "Commodity Portfolio"
11        }
12      }
13     accounting_date: {
14       gte: "2021-01-01"
15       lte: "2023-12-31"
16     }
17   }
18 } {
19   id
20   accounting_date
}
```

```
{
  "data": {
    "transaction": [
      {
        "id": "81",
        "accounting_date": "2023-06-01",
        "position_size_with_sign": 2000,
        "cash_consideration": -2457770,
        "trade_price": 1228.89,
        "instrument": {
          "id": "30",
          "user_code": "XPD",
          "name": "XPD - Palladium (Commodity), USD (XPD)"
        },
        "portfolio": {
          "id": "13",
          "user_code": "Commodity Portfolio",
          "name": "Commodity Portfolio"
        }
      }
    ]
  }
}
```

GraphQL Query

```
query GetTransactionList {
  transaction(
    pagination: {
      limit: 20
      offset: 0
    }
    filters: {
      portfolio: {
        user_code: {
          exact: "Commodity Portfolio"
        }
      }
      accounting_date: {
        gte: "2021-01-01"
        lte: "2023-12-31"
      }
    }
  ) {
    id
    accounting_date
  }
}
```

```
) {
  id
  accounting_date
  position_size_with_sign
  cash_consideration
  trade_price
  instrument {
    id
    user_code
    name
  }
  portfolio {
    id
    user_code
    name
  }
  account_cash {
    id
    user_code
    name
  }
  account_position {
    id
    user_code
    name
  }
  transaction_class {
    id
    user_code
    name
  }
}
}
```

Filter Highlights

Filter by Portfolio (nested filter)

```
portfolio: {  
  user_code: { exact: "Commodity Portfolio" }  
}
```

- `portfolio` is a related object.
- You filter it by its field `user_code`.
- `exact` means strict match.

Filter by Date Range

```
accounting_date: {  
  gte: "2021-01-01"  
  lte: "2023-12-31"  
}
```

- `gte` = start date (inclusive)
- `lte` = end date (inclusive)
- Use ISO format: `YYYY-MM-DD`

How filters work together

All filters in one `filters` block are combined with **AND**.

So this query means:

- portfolio `user_code` is exactly `"Commodity Portfolio"`
- **AND**
- `accounting_date` is between `2021-01-01` and `2023-12-31`

Revision #3

Created 2025-12-16 16:51:48 UTC by Sergei Zhitenev

Updated 2025-12-16 16:58:40 UTC by Sergei Zhitenev