ETL TRANSACTION API USER GUIDE

This guide shows how to use the API, which is available through the Swagger interface.

You can open the Swagger UI here to try out the endpoints:

https://yokomitsu.ddns.net:5444/swagger/index.html

If you prefer using a tool like Postman, you can also access the API directly at:

https://yokomitsu.ddns.net:5444/api/<endpoint>

1.0 Get Access Token

1.1 Retrieving the Access Token:

To begin, go to the **"Tokens"** section in the Swagger UI. There, you'll find the endpoint

/api/getAccessToken to retrieve an access token for authentication and security purposes.

You must include the following parameters:

- **companyName**: This is required for generating the token. Make sure to include it as a parameter (as shown in diagram 1.1).
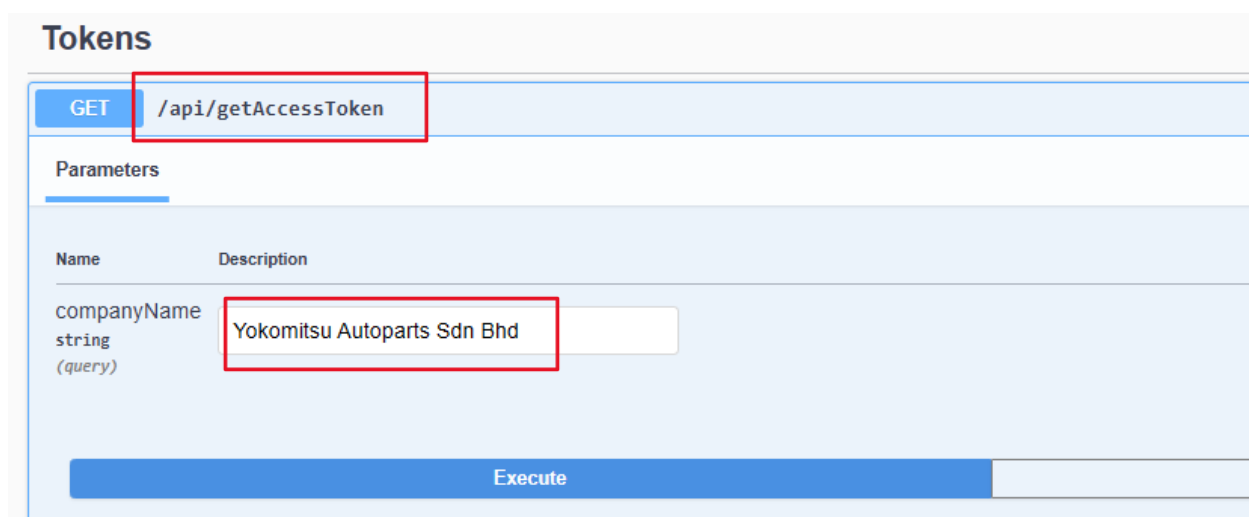

Diagram 1.1 Parameter for Access Token Retrieval

After submitting the company name, you will receive an access token that looks like this:

tjIo5P6plhrmESxhHX0ELu6ixvoooTLIB6XALzjx2K4=

Server response

Code   Details

200
            Response body

            tjIo5P6plhrmESxhHX0ELu6ixvoooTLIB6XALzjx2K4=

Diagram 1.2 Response Structure for Access Token API

2.0 Get Sales Transactions

2.1 Retrieving Sales Transactions:

To begin, go to the "**SalesTransactions**" section in the Swagger UI. There, you'll find the endpoint **/api/sales/transactions** to retrieve a list of transaction records.

You must include the following parameters:

- **accessToken**: This is required for authentication. Make sure to include it as a parameter (as shown in diagram 2.1).

In the **request body**, provide:

- **startDate** and **endDate**: Specify the date range in ISO format.
- **page**: Page number for pagination **(must be greater than 0)**.
- **limit**: Number of records per page **(must be greater than 0)**.

Diagram 2.1 Parameter and Request Body for Sales Transactions API

Server response

Code   Details

200
           Response body
           {
             "totalCount": 44,
             "page": 1,
             "limit": 10,
             "message": "The data is too large. Only partial results are shown. Please increase the page number to see more.",
             "results": [
               {
                 "documentNo": "IV-05125",
                 "transactionType": "INV",
                 "transactionDescription": "SALES",
                 "stockCode": "71411512072",
                 "quantity": 1,
                 "totalAmount": 59,
                 "unitCost": 0,
                 "location": "L1",
                 "issueDate": "2025-05-19T00:00:00",
                 "customerCode": "300-Z0001",
                 "customerName": "ZHAO FENG SDN. BHD.",
                 "salesmanCode": "JAYDEN",
                 "salesmanName": "JAYDEN",
                 "nettQty": 37
               },
               {
                 "documentNo": "IV-05126",
                 "transactionType": "INV",
                 "transactionDescription": "SALES",
                 "stockCode": "71552661080",

Download

           Response headers
           content-type: application/json; charset=utf-8
           date: Mon,19 May 2025 02:22:18 GMT
           server: Microsoft-IIS/10.0
           x-powered-by: ASP.NET

Diagram 2.2 Response Structure for Sales Transactions API

Note:

- You can set the limit value up to **200**.

- Feel free to increase the page number to retrieve more transactions from the next set of results.

3.0 Get Current Inventories

3.1 Retrieving Current Inventories:

To begin, go to the " **InventoryQuantity**" section in the Swagger UI. There, you'll find the endpoint **/api/inventory/currentInventory** to retrieve a list of current inventories records or just a single inventory record.

You must include the following parameters:

- **accessToken**: This is required for authentication. Make sure to include it as a parameter (as shown in diagram 3.1).

In the **request body**, provide:

- **stockCode**:
    1. To retrieve a specific stock record, enter the desired stock code (e.g., " **31231211141**"). In this case, you can leave both **page** and **limit** as 0 (as shown in diagram 3.2).
    2. To retrieve all stock records at once, leave **stockCode** as an empty string "" and leave **page** and **limit** as 0 (both must be greater than 0 for pagination) (as shown in diagram 3.3).
    3. To retrieve stock records in pages, leave **stockCode** as an empty string "", but provide valid **page** and **limit** values (both must be greater than 0) to enable pagination (as shown in diagram 3.4).
- **page**: Page number for pagination **(must be greater than 0)**.
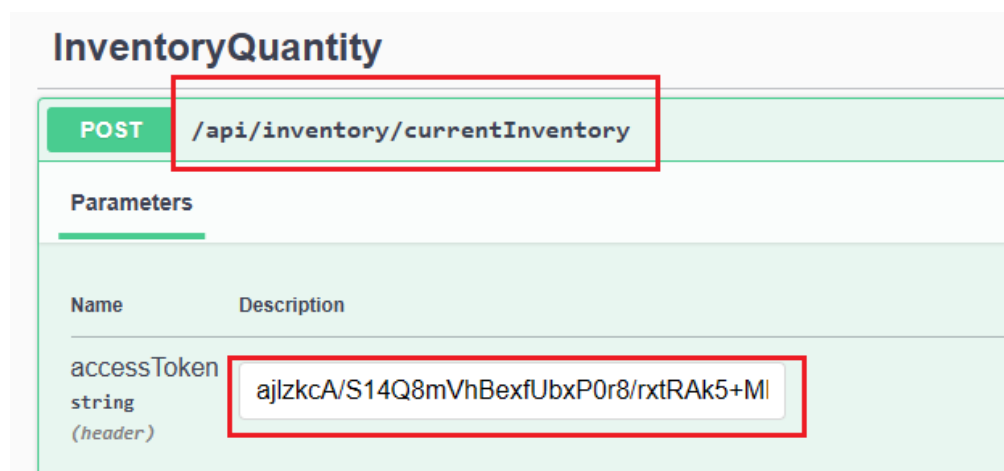- **limit**: Number of records per page **(must be greater than 0)**.

Diagram 3.1 Parameter for Inventory Quantity API

Request body

```
{
"stockCode": "31231211141",
"page": 0,
"limit": 0
}
```

Server response

Code	Details
200	**Response body**

```
{
  "count": 1,
  "page": 0,
  "limit": 0,
  "message": "Inventories fetched successfully.",
  "result": [
    {
      "stockCode": "31231211141",
      "stockName": "FENDER RH W/O H CITY 2014",
      "locations": [
        {
          "locationCode": "L1",
          "quantity": 16
        },
        {
          "locationCode": "L2"
```

Diagram 3.2 Request and response for retrieving a specific inventory record

Request body

```json
{
  "stockCode": "",
  "page": 0,
  "limit": 0
}
```

Server response

Code	Details
200	**Response body**

```json
{
  "count": 1019,
  "page": 0,
  "limit": 0,
  "message": "Inventories fetched successfully.",
  "result": [
    {
      "stockCode": "*",
      "stockName": "NON-STOCK",
      "locations": [
        {
          "locationCode": "L1",
          "quantity": 100
        },
        {
          "locationCode": "L2",
          "quantity": 0
        },
        {
          "locationCode": "L3",
          "quantity": 0
        },
        {
          "locationCode": "L4",
          "quantity": 0
        },
        {
          "locationCode": "L5",
```
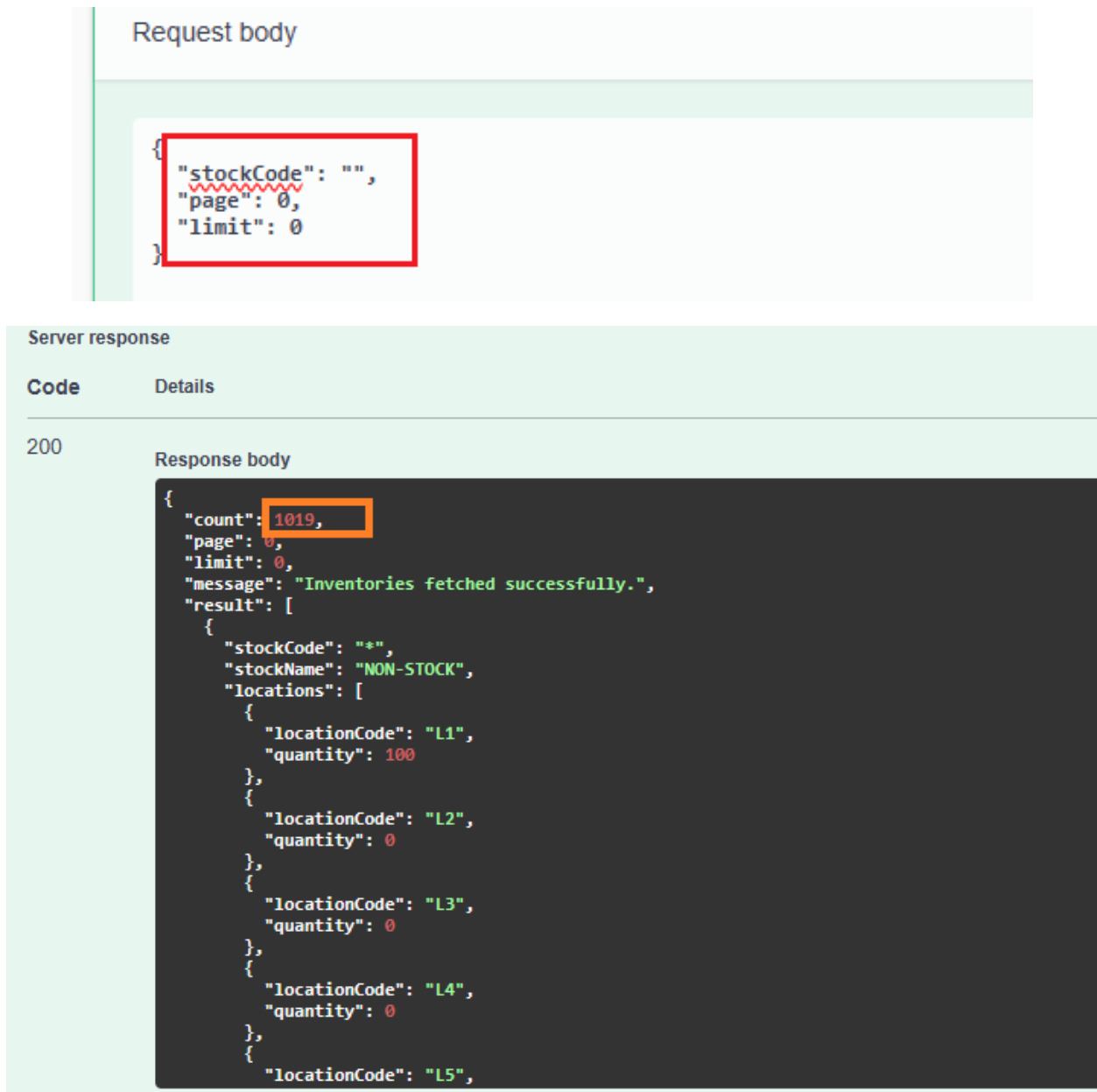
Diagram 3.3 Request and response for retrieving all inventory records

Request body

```
{
"stockCode": "",
"page": 1,
"limit": 10
}
```

Server response

Code	Details
200	Response body

```
{
   "count": 1019,
   "page": 1,
   "limit": 10,
   "message": "Inventories fetched successfully.",
   "result": [
      {
         "stockCode": "*",
         "stockName": "NON-STOCK",
         "locations": [
            {
               "locationCode": "L1",
               "quantity": 100
            },
            {
               "locationCode": "L2",
               "quantity": 0
            },
            {
               "locationCode": "L3",
               "quantity": 0
            },
            {
               "locationCode": "L4",
               "quantity": 0
            },
            {
               "locationCode": "L5"
```

Diagram 3.4 Request and response for retrieving inventory records with pagination (all stock records within a specific range)

Note:

- If you leave stockCode as an empty string "" and set both page and limit to 0, the API will retrieve all inventory records at once. While this is allowed, it is not recommended due to potential performance issues or long response times with large datasets.

- For better performance, use pagination by setting page and limit to values greater than 0.

- You can set the limit value up to **200**.

- Feel free to increase the page number to retrieve more inventory records from the next set of results.