

API implementation guide and integration overview

Date : 08-07-2024
Version : 1.12

Index

Changelog	3
How to integrate your website or IT system with the midocean IT systems.	4
Who can use the API?	4
How can I request a Rest API key?	4
Note about Test environment!	5
What information files does API offer?	5
What services does API offer?	6
API Information Files	7
Stock Information 2.0	7
Product information 2.0	8
Print Data 1.0	10
API Price files	13
Product Pricelist 2.0	13
Print Pricelist 2.0	14
API Order Services	17
Order Entry 2.1	17
Fields Appendix	17
Order Request format (JSON)	20
Order Response	23
Error Resolutions	24
Order Details 2.1	25
Approve Proof 1.0	27
Reject Proof 1.0	28
Add Artwork 1.0	28
APPENDIX A: Rest API Integration using Postman	30
APPENDIX B: How to Download and Import Postman Collection from Customer API Page JSON or URL	33
Postman Collection: JSON	33
Postman Collection: URL	36
APPENDIX C: API Integration Errors	38
APPENDIX D: Printed API Order Flow	39
APPENDIX E: List of rejection reasons	39

Changelog

- V.1.12
 - Update of order entry endpoint (new version)
 - Update of order details endpoint (additional values)
 - Addition of new Proof management APIs.
 - Proof approval API.
 - Proof rejection API.
 - List of rejection reasons
 - Add artwork API.
 - Postman collection is updated to include the new APIs.
 - Added Appendix D: Print Order API flow diagram.
- V.1.11
 - Order Entry
 - Specified mandatory fields.
 - Order Entry
 - Added/Expanded fields explanations.
 - Removed SFTP references completely (SFTP service is now deprecated and will be discontinued in 2025).
 - Customer API page rework.
 - Adjusted API key retrieval instructions.
- V.1.10.1
 - Compacted SFTP file information.
 - Remade code snippets.
- V.1.10
 - Rework of the Rest API method of request.
 - Added entry about the new Customer API page in the webshop.
- V.1.9
 - Complete removal of Webservices API references since it's now obsolete.
 - Addition of Order Entry Rest API.
 - Addition of Order Details & Tracking Rest API.
 - Rework of the whole layout.
 - Added SFTP and Rest API setup guides.
 - Added API error appendix.
- V.1.8
 - Added Stock API Call.
 - Added Print Data API Call.

How to integrate your website or IT system with the midocean IT systems.

Who can use the API?

The midocean API can be used by Distributors that have their own Website or IT system. By integrating our API, you will get accurate product, stock, pricing and print configuration data to your End Users, without having to perform any manual procedure (everything will run automatically).

In addition, the midocean API can be used to place orders, check order details, and order tracking.

How can I request a Rest API key?

Be aware that to make connections with the midocean systems you will require an API key to authorize your connections with our Rest API environment.

Firstly, you need to have an account on the website <https://www.midocean.com>

- Log in to your midocean account, go to your Account, then access the Customer API tab.
- Click Register to create an API key that's linked to your midocean account.

API key

In order to access midocean's REST APIs, you would need an API key.

Below you can request/view such a key, which can be used by your developers to obtain access to the REST APIs.

Once you have your API key, you'll need to subscribe to the specific API or API(s) you want to use.

Simply tick the boxes next to the desired API/APIs and click "Update" to activate them.

Register

- Page will refresh and your API key will be displayed.

API key

In order to access midocean's REST APIs, you would need an API key.

Below you can request/view such a key, which can be used by your developers to obtain access to the REST APIs.

Once you have your API key, you'll need to subscribe to the specific API or API(s) you want to use.

Simply tick the boxes next to the desired API/APIs and click "Update" to activate them.

Your API Key: XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX

- Now you can choose which endpoint(s) to subscribe to by selecting them and clicking on Update

Subject	Description	Endpoint
✓ Printpricelist	The print pricelist Rest API can be used to retrieve print prices in JSON (Accept header = application/json) or XML (Accept header = text/xml) or CSV (Accept header = text/csv) format.	https://api.midocean.com/gateway/printpricelist/2.0
✓ Pricelist	The pricelist Rest API can be used to retrieve the product prices in JSON (Accept header = application/json), XML (Accept header = text/xml) or CSV (Accept header = text/csv) format.	https://api.midocean.com/gateway/pricelist/2.0
✓ Printdata	The printdata Rest API can be used to retrieve print information in JSON (Accept header = application/json) or XML (Accept header = text/xml) format.	https://api.midocean.com/gateway/printdata/1.0
✓ Order	The order Rest API can be used to create an order (method: /create) or retrieve order detail/tracking (method: /detail) in JSON (Accept header = application/json) or XML (Accept header = text/xml) format.	https://api.midocean.com/gateway/order/2.0
✓ Stock	The stock Rest API can be used to retrieve the current and upcoming stock levels in JSON (Accept header = application/json), XML (Accept header = text/xml) and CSV (Accept header = text/csv) format.	https://api.midocean.com/gateway/stock/2.0
✓ Products	The products Rest API can be used to retrieve information about products in JSON (Accept header = application/json), XML (Accept header = text/xml) or CSV (Accept header = text/csv) format.	https://api.midocean.com/gateway/products/2.0

Update

Note about Test environment!

There are two Rest API environments.

Each of the two environments requires a different key and a different URL header.

- **Production:**
 - This is the normal environment, the calls made here are reflected on our live systems and on the website, so real data and real orders.
- **Test:**
 - Calls made in the Test Environment are just for testing and to familiarize yourself with the API system, you can use the Test environment to practice with the Order Create Rest API without any risk.

To request the API Key for Test please contact your regular midocean salesperson and ask for a Rest API key for Test environment specifically.

To make use of our TEST environment, replace the URL header from <https://api.midocean.com> to <https://apitest.midocean.com> and replace your production API key with your test API key

What information files does API offer?

Using the Rest API you can integrate almost all data relating to products, stock, printing and the prices you have assigned from the website, the files available are:

- **Stock Information 2.0**
 - <https://api.midocean.com/gateway/stock/2.0>

With this file you can upload in your system the current stock levels of our entire stock product assortment.

Available in json, xml and csv.

Updated each hour.

- **Product Information 2.0**

- <https://api.midocean.com/gateway/products/2.0?language=en>

Contains all product data that midocean can share like descriptions, images, attributes, documents, etc. You can upload in your system all the product information, including textiles, that we have available in our Webshop.

NOTE: You can request the file in most European languages by changing the language key in the endpoint URL, "en" is for English, but if requesting in Spanish the key is "es" instead on "en"

Available in multiple languages: en, es, pt, fr, it, nl, de, hu, ro, cs, sv, fi, da.
Available in json, xml and csv.
Updated daily.

- **Product Pricelist 2.0**

- <https://api.midocean.com/gateway/pricelist/2.0/>

This file contains the specific customer pricing per item (any discount or condition that you have will be reflected on this file).

The price that you see will be the same prices that you can see on the midocean Webshop or the one that will be given by the Customer Care.

Available in json, xml and csv.
Updated daily.

- **Print Pricelist 2.0**

- <https://api.midocean.com/gateway/printpricelist/2.0/>

This file contains the print prices per print technique and handling costs per item.

The price that you see will be the same prices that you can see on the midocean Webshop or the one that will be given by the Customer Care.

Available in json, xml and csv.
Updated daily.

- **Print Data 1.0**

- <https://api.midocean.com/gateway/printdata/1.0>

By integrating this file you can upload in your system all the Printing options that are available for the midocean products, print positions, available printing techniques per position, the max print size, max print colors and template images of the positions.

Available in json and xml.
Updated daily.

What services does API offer?

We also offer some API services that allow customers to create and manage their orders from their own systems without the need to access the Webshop or call Customer Care at all!

From version 1.11 onwards of the documentation we also give access to some APIs that allow you to manage the proof status of print orders, so you can even Approve, Reject or Add artwork to printed orders.

Note that these calls require more configuration to setup, since they can be personalized for each case. If you're getting familiarized with the API, we recommend to [follow the instructions to import the Postman collection](#) which automates the configuration of most of the setup.

The services available via Rest API are:

- **Order Entry**

- <https://api.midocean.com/gateway/order/2.0/create>

This API will allow you to place orders to midocean without the need to use the Webshop or call Customer Care.

To make use of this API you will need to send via POST a request json file that will contain all the order details, once the details are sent the system will return the order confirmation along with the order number and pricing details.

You can integrate this functionality in your ERP system or directly on your website, so when a customer places an order the order is transferred directly to midocean.

- **Order Details**

- https://api.midocean.com/gateway/order/2.1/detail?order_number=order_number

This API will return the fully updated order details for any of your orders placed in our midocean systems, the updated tracking information for your orders, including forwarder ID, the tracking link and in case of print orders the latest proof download link.

As of the 2.1 version of this call the proof status for printed orders is also available so you can check if proof is ready or if there is an objection on the artwork provided. The list of proof statuses is explained in the [Order details section](#)

- **Proof Approval
(Print orders)**

- <https://api.midocean.com/gateway/proof/1.0/approve>

This API will send an approve request on any order you placed in our midocean systems so you can approve proofs without even having to visit the website.

- **Proof Rejection
(Print orders)**

- <https://api.midocean.com/gateway/proof/1.0/reject>

This API can be used to reject a proof for a printed order, you can send a request with the order number, order line, rejection reason (see list of rejection reasons) and any additional files or new artwork that you want to upload.

- **Add Artwork
(Print orders)**

- <https://api.midocean.com/gateway/proof/1.0/addartwork>

If your printed order gets a rejection because of the artwork, you can upload new artwork by sending a request with this API.

API Information Files

Stock Information 2.0

The stock information file includes the available stock of all products available on the website.

The **Stock Information** Rest API has to be called at the following URL by making a GET call:

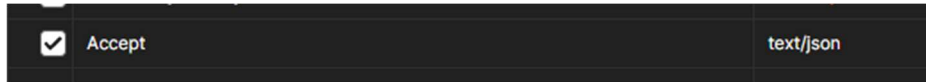
- <https://api.midocean.com/gateway/stock/2.0>

- Stock information file is updated **every hour**
 - Actual Stock shown is the aggregate of all warehouses
 - Stock is the actual and the next two arrivals that have been shipped from China
 - All midocean Products that are on the Webshop are on the BC Stock

NOTE:

Available in JSON, XML and CSV formats, by default output is in JSON

Add a header with the key **"Accept"** and set the value depending on the wanted format, **text/"format"** (**text/json**, **text/xml**, **text/csv**)



An example could look as follows. Note that this example only shows a small part of the result.

The header of the stock file indicates the last update time, since the stock information is updated hourly this is reflected on the header

```
{
  "modified_at": "2023-08-11T23:45:09+02:00",
  "stock": {
```

The stock API returns the available stock in our warehouse and the next two stock arrivals if applicable. In this case it tells us that item KC2364-37 does not have current stock, but on 25/08/2023 3000 pcs will arrive to the Warehouse and also another shipment will arrive on 22/09/2023 with 5000 extra pcs.

```
{
  "sku": "KC2364-37",
  "qty": 811,
  "first_arrival_date": "2023-08-25",
  "first_arrival_qty": 3000,
  "next_arrival_date": "2023-09-22",
  "next_arrival_qty": 5000
},
```

Product information 2.0

The new product information version 2.0 file is only available as a REST API.

To make things easier we're also providing a CSV version on the SFTP server, based on the REST API.

At this moment it isn't possible to retrieve only a portion of the product data. This functionality will also be added during late 2024. That will be useful to only retrieve products that have changed since a certain date.

The **Product Information** Rest API has to be called at the following URL by making a GET call:

- <https://api.midocean.com/gateway/products/2.0?language=en>

Note that there's a parameter added to the url that determines the language in which you'll receive the product data (**?language=en**). Available languages are (English, Czech, German, Spanish, French, Hungarian, Italian, Dutch, Polish, Portuguese, Romanian, Russian, Swedish)

NOTE:

Available in JSON, XML and CSV formats, by default output is in JSON

Add a header with the key **"Accept"** and set the value depending on the wanted format, **text/"format"** (**text/json**, **text/xml**, **text/csv**)

<input checked="" type="checkbox"/> Accept	text/json
--	-----------

Here's an example of the response in json for product AR1804.

The response includes all the details, like the product code, master ID, category, and a list of all available attributes (depends on product)

```
{
  "master_code": "AR1804",
  "master_id": "40000190",
  "type_of_products": "stock",
  "commodity_code": "4820 1030",
  "number_of_print_positions": "5",
  "country_of_origin": "CN",
  "brand": "midocean",
  "product_name": "ARCONOT",
  "category_code": "MOBOFF_NOTOTH",
  "product_class": "Notebooks & notepads",
  "dimensions": "21X14X1,6 CM",
}
```

Next you can find a list of all included documents, certifications and compliancy documents for the product in PDF format.

```
{
  "digital_assets": [
    {
      "url": "https://cdn1.midocean.com/document/declaration-of-sustainability/ar1804-green.pdf",
      "type": "document",
      "subtype": "declaration_of_sustainability"
    },
    {
      "url": "https://cdn1.midocean.com/document/declaration-of-conformity/ar1804-doc.pdf",
      "type": "document",
      "subtype": "declaration_of_conformity"
    },
    {
      "url": "https://cdn1.midocean.com/document/test-reach/ar1804-test-reach.pdf",
      "type": "document",
      "subtype": "test_reach"
    }
  ]
}
```

The short description and the long description, some more information about material and if it can be printed or not.

```
{
  "short_description": "A5 notebook 96 plain sheets",
  "long_description": "A5 notebook with hard PU cover. Casebound. 192 plain pages (96 sheets). Matching elastic closure strap and ribbon page-marker.",
  "material": "PU",
  "green": "yes",
  "printable": "yes",
  "polybag": "no",
}
```

After this section all the variants of the product are listed, **a product will have one or multiple variants, containing colours and/or sizes. Images are available on variant level**

```
"variant_id": "10168709",
  "sku": "AR1804-03",
  "release_date": "2015-01-01",
  "product_proposition_category": "295",
  "category_level1": "Office & Writing",
  "category_level2": "Notebooks",
  "category_level3": "Hard cover",
  "color_description": "black",
  "color_group": "black",
  "plc_status": "16",
  "plc_status_description": "COLLECTION",
  "gtin": "8719941012868",
  "color_code": "03",
  "pms_color": "BLACK",
```

```
"digital_assets": [
  {
    "url": "https://cdn1.midocean.com/image/700X700/ar1804-03.jpg",
    "url_highress": "https://cdn1.midocean.com/image/original/ar1804-03.jpg",
    "type": "image",
    "subtype": "item_picture_front"
  },
  {
    "url": "https://cdn1.midocean.com/image/700X700/ar1804-03-back.jpg",
    "url_highress": "https://cdn1.midocean.com/image/original/ar1804-03-back.jpg",
    "type": "image",
    "subtype": "item_picture_back"
  }
]
```

Print Data 1.0

The Print data file contains the specifics of printing of each single item in our collection, every time that there is a change on the printing information of a product in our system the file is updated overnight. Printing Data has the following features:

- Can be requested via REST API, in json and xml format.
- Includes Printing Information about Textiles
- Product variants are not separated since they have the same printing information.
- Shows specifics print area points.

The **Print Data** Rest API has to be called at the following URL by making a GET call:

- <https://api.midocean.com/gateway/printdata/1.0>

NOTE:

Available in JSON and XML formats, by default output is in JSON.

Add a header with the key **"Accept"** and set the value depending on the wanted format, **text/"format"** (**text/json**, **text/xml**)

<input checked="" type="checkbox"/>	Accept	text/json
-------------------------------------	--------	-----------

The printdata file starts with a set of labels with the translations of each Printing Technique in the different languages, so you will not need to make the translations as they can be picked up directly from the file.

```
<printing_technique_descriptions>
  <printing_technique_description>
    <id>B</id>
    <name>
      <cs>Ražba</cs>
      <da />
      <de>Prägung</de>
      <en>Debossing</en>
      <es>Termo grabado</es>
      <fi>Kohokuviointi</fi>
      <fr>Embossage</fr>
      <hu>Dombornyomás</hu>
      <it>Debossing</it>
      <nl>Blinddruk</nl>
      <pl>Tłoczenie</pl>
      <pt>Impressão à quente</pt>
      <ro>Debossing</ro>
      <ru>Тиснение</ru>
      <sv>Prägling</sv>
    </name>
  </printing_technique_description>
</printing_technique_descriptions>
```

Then, it lists item color variations by code per item.

```
<product>
  <master_code>AR1804</master_code>
  <master_id>40000190</master_id>
  <item_color_numbers>
    <item_color_number>03</item_color_number>
    <item_color_number>04</item_color_number>
    <item_color_number>05</item_color_number>
    <item_color_number>06</item_color_number>
    <item_color_number>10</item_color_number>
    <item_color_number>48</item_color_number>
    <item_color_number>12</item_color_number>
    <item_color_number>37</item_color_number>
  </item_color_numbers>
```

After that information you will get an specific data about the printing options per item. Please, notice that in this case the relation is N:1 as the printing information is the same for the different colours of an item. In this case you can relate the Product Information file with the Printing Information file by the label “master_code”.

The tag “print_manipulation” specifies the manipulation group of an item, it can be **A, B, C, D, E or Z** (in some cases if there is none the code will show as null)

If you search this letter code in the print pricelist file you can get the prices for the handling of a product.

```
<print_manipulation>Z</print_manipulation>
<print_template>https://printtemplates-v2.cdn.midocean.com/AR1804-print-
template.pdf</print_template>
```

The tags “print_template” contain a pdf with the different print position images which show the layout of the print area, sizes and technique codes.

Product

AR1804

Handling code: Z

<div>Print position</div> <ol style="list-style-type: none"> 1. FRONT 2. FRONT PD 3. BACK 4. BACK PD 5. FRONT PAD 	<div>Print size</div> <p>120,0 mm x 190,0 mm 147,0 mm x 210,0 mm 90,0 mm x 190,0 mm 111,0 mm x 210,0 mm 50,0 mm x 30,0 mm</p>	<div>Print code (no of colours)</div> <p>B(1),S2(4),T1(8),TD(8) PD4(8) B(1),S2(4),T1(8),TD(8) PD4(8) P3(2)</p>
--	---	--

1

2

3

In case that there are several Print Positions in one item, you will see that reflecting on the file as different "position_id" will be shown. Each one of them will have their specific Printing Techniques, Max. Print Size and Max. Print Colours.

```

<printing_position>
  <position_id>FRONT</position_id>
  <print_size_unit>mm</print_size_unit>
  <max_print_size_height>190.00</max_print_size_height>
  <max_print_size_width>120.00</max_print_size_width>
  <rotation>0.00</rotation>
  <print_position_type>Rectangle</print_position_type>
  <printing_techniques>
    <printing_technique>
      <default>>false</default>
      <id>TD</id>
      <max_colours>0</max_colours>
    </printing_technique>
    <printing_technique>
      <default>>false</default>
      <id>B</id>
      <max_colours>1</max_colours>
    </printing_technique>
    <printing_technique>
      <default>>false</default>
      <id>T1</id>
      <max_colours>8</max_colours>
    </printing_technique>
    <printing_technique>
      <default>true</default>
      <id>S2</id>
      <max_colours>4</max_colours>
    </printing_technique>
    <printing_technique>
      <default>>false</default>
      <id>TR</id>
      <max_colours>1</max_colours>
    </printing_technique>
  </printing_techniques>

```

This file also includes a <points> tag, it indicates the exact location of the printing points in pixels.

```
<point>
  <distance_from_left>67</distance_from_left>
  <distance_from_top>45</distance_from_top>
  <sequence_no>1</sequence_no>
</point>
<point>
  <distance_from_left>632</distance_from_left>
  <distance_from_top>940</distance_from_top>
  <sequence_no>2</sequence_no>
</point>
```

Then it links to the images used in each print position, one with printable area drawn and the other without.



API Price files

Product Pricelist 2.0

The output will be a complete file with the specific customer pricing (**based on the api key on the input**) for the entire midocean Product assortment.

You will have to integrate the content of the output json file into your database.

The **Print Pricelist** API has to be called at the following URL by making a GET call:

- <https://api.midocean.com/gateway/pricelist/2.0>

Your API key needs to be added to the header of the GET call.

NOTE: The output is by default a .json file.

To change the output format to xml add a header with the key "**Accept**" and set the value to:

text/json - request in .json or

text/xml - request in .xml

The first part of an example json output will look as follows:

```
{
  "currency": "EUR",
  "date": "2023-10-31",
  "price": [
    {
      "sku": "AR1249-16",
      "variant_id": "10134325",
      "price": "4,22",
      "valid_until": "2024-01-31",
      "scale": [
        {
          "minimum_quantity": "1000",
          "price": "3,77"
        },
        {
          "minimum_quantity": "500",
          "price": "3,92"
        },
        {
          "minimum_quantity": "250",
          "price": "4,07"
        },
        {
          "minimum_quantity": "1",
          "price": "4,22"
        }
      ]
    }
  ],
}
```

Note that tier prices might or might not be available, depending on your region and account.

Print Pricelist 2.0

The Print Pricelist Rest API will return your specific midocean Print Prices. You can use the prices included to make your own printed orders using our Order Entry API.

The **Print Pricelist** Rest API has to be called at the following URL by making a GET call:

- <https://api.midocean.com/gateway/printpricelist/2.0>

NOTE:

Available in JSON, XML and CSV formats, by default output is in JSON

Add a header with the key **"Accept"** and set the value depending on the wanted format, **text/"format"** (**text/json**, **text/xml**, **text/csv**)

<input checked="" type="checkbox"/>	Accept	text/json
-------------------------------------	--------	-----------

Sample response:

The print pricelist starts with a header that shows the pricelist currency, validity date. After that there's a section with the different print manipulations prices and codes to relate to the printing data information.

```
{
  "currency": "EUR",
  "pricelist_valid_from": "2023-10-31",
  "pricelist_valid_until": "2023-12-15",
  "print_manipulations": [
    {
```

Next it shows the manipulation (handling) prices based on the code available in the [Print Data](#) in the field "print_manipulation"

```
{
  "code": "A",
  "description": "Simple",
  "price": "0,08"
},
```

The printing specific section, here you can check the ID based on info in [Print Data](#) file, the pricing type, which defines the way that the prices are calculated for the print technique, calculation types can be found in [Appendix B](#)

```
"print_techniques": [
  {
    "id": "B",
    "description": "Embossing",
    "pricing_type": "NumberOfPositions",
    "setup": "60,00",
    "setup_repeat": "20,00",
    "next_colour_cost_indicator": "false",
    "var_costs": [
      {
        "range_id": "",
        "area_from": "0",
        "area_to": "0",
        "scales": [
          {
            "minimum_quantity": "1",
            "price": "1,56",
            "next_price": ""
          },
          {
            "minimum_quantity": "50",
            "price": "0,94",
            "next_price": ""
          },
          {
            "minimum_quantity": "100",
            "price": "0,81",
            "next_price": ""
          }
        ]
      }
    ]
  }
],
```

General explanation of Scales: Scales are based on the item quantity. If you find the scales 1, 50, 100, ... in the file and your item quantity is 60, you have to use the price of scale 50.

General explanation of Manipulation cost: In your calculation you have to multiply the item quantity with the corresponding cost of the manipulation for that item.

The way print prices need to be calculated is described in the former chapter.

Please also take advantage of the calculation examples available in [Appendix B](#).

Pricing Type	Price calculation	Example calculation
Number of positions	Print prices are based on the number of positions that you want to print. For example print technique B (Embossing).	Price calculation in case of 3 positions = scale-prices (depending on the quantity) * 3 positions * item-quantity.
Number of colours	Print prices are based on the number of colors of the logo. If next_colour_cost_indicator='X' than the 2nd and next colors have another price (price_next)	Price calculation in case of a 4 colors logo with next_colour_cost_indicator=' ' = price_1st * 4 * item-quantity Price calculation in case of a 4 colors logo with next_colour_cost_indicator='X' = price_1st * 1 * item-quantity + price_next * 3 * item quantity
AreaRange	Print prices are assigned depending on the print area to be used (surface / cm2 of the logo to be printed). There are different price ranges depending on the area. Once the range is located you can use the price assigned (there is no need to multiply it by the logo area size). See example of technique E: <pre> <printing_technique> <id>E</id> <description>Embroidery</description> <pricing_type>AreaRange</pricing_type> <setup>36,00</setup> <setup_repeat>20,00</setup_repeat> <apply_cost_ind/> <apply_cost/> <next_colour_cost_indicator/> <var_costs> <var_cost> <range_id>A</range_id> <area_from>0</area_from> <area_to>8</area_to> <scales> <scale> <qty_min>1</qty_min> <price_1st>2,162</price_1st> <price_next/> </scale> </scales> </var_cost> </var_costs> </printing_technique> </pre>	Price calculation in case of a 6 cm2 Logo with print technique Embroidery = 2,162 * item-quantity
ColourAreaRange	Print prices are based on the combination of Area ranges and colors of the logo. (Selection of area range depending on the area and price multiplied by the number of colors of the logo).	print-price (selection of range based on area cm2) * number-of-colors * item-quantity and price multiplied by the number of colors of the logo).

Special calculation for Printing Technique id ST: For this technique, there is a difference in the calculation between white items and colored items. In the calculation you have to add 1x price_next * item-quantity extra in case of colored items. The reason behind this extra cost for non-white items is that an extra white layer needs to be printed before printing the actual logo to get the right quality and expected result.

You can find [Appendix B](#) with a table that contains a sample calculation for each pricing type.

API Order Services

Order Entry 2.1

By using this API the Distributors of midocean will be able to place orders directly to the midocean SAP Logistics System. Placing an order using this method will have the same response as of placing it through our Customer Care Agents or by placing the order on the midocean Webshop.

The types of orders that can be placed using this Service are:

- **Normal Orders**
 - Unprinted with no quantity limit
- **Print Orders**
 - Printed with no quantity limit
- **Sample Orders**
 - Unprinted with 5 piece per line limit

NOTE: It's not possible to combine different types of orders in the same order request.

Own Design method:

Depending on your expertise with the creation and design of proofs, you can arrange with your Sales Manager to directly be able to submit your confirmed artworks without having to go through the process of validation, waiting for proof creation and having to check the proofs, which will make the whole process faster and easier.

Once the order is placed in the midocean system the only way to cancel it or modify it is by direct contact with our Customer Care Agents.

Notice that to make use of this service the following files need to be integrated in your system:

- **Normal/Sample orders:**
 - [Stock Information](#)
 - [Product Information](#)
 - [Pricing Information](#)
- **Print orders:**
 - [Stock Information](#)
 - [Product Information](#)
 - [Printing Information](#)
 - [Pricing Information](#)
 - [Print Pricing Information](#)

Just once the above mentioned files are integrated; it is then possible to make full use of the Order Entry Rest API

Before explain the format of the Input for the Order Entry 2.1, it is important that in order to minimise errors a **pre-stock check tool is used on the Distributor Website or ERP** to avoid placing orders for which the stock cannot be fulfilled. This can be simply done by integrating the Stock file and doing a simple check at the order entry stage.

The **Order Entry** Rest API has to be called at the following URL by making a **POST** call:

- <https://api.midocean.com/gateway/order/2.1/create>

Fields Appendix

Here's a list of the fields that are specified in the Order Entry 2.0 POST call along with explanations. Mandatory fields need to have values, (e.g. printing_technique_id: "**S2**",) If a field is not specified to be mandatory it can have an empty value, (e.g. po_number: "",)

Regular Fields

order_header

- **check_price**
Can be set to true or false.
 - If set to true, the system will check the price specified in "expected_price" and if there's a mismatch with the prices calculated by our systems the order entry is cancelled.
 - If set to false, the system automatically applies the price based on the product information provided and doesn't consider the "expected_price".
- **timestamp**
YYYY-MM-DD followed by a T and the HH:MM:SS. (e.g. **2023-03-20T08:17:00**).
- **currency**
Your prices currency
- **po_number:**
Here you can enter a purchase order number in order to reference your order.
Note that the API blocks the order entry if it detects that you use the same po_number for a different order as to avoid duplicated orders.
- **contact_name**
- **contact_email**

shipping_address

- **order_type**
Choose if the order is unprinted "**NORMAL**", a sample "**SAMPLE**" or has printing "**PRINT**"
- **preferred_shipping_date**
If you prefer to have the order delivered on a specific date you can specify it here. If set to a previous date than the order the field won't be used and the order will be delivered always as soon as possible. Optional field, if left empty the first system date will be used
YYYY-MM-DD | **Ex: 2023-03-20**
- **country**
Set the shipping destination country code, i.e. ES, FR, NL, etc.

NOTE: Depending on your account's region the number of countries you can ship to is limited. For more information on which countries are allowed for your sales organization please contact your regular midocean salesperson.

The list of countries is 1 to 1 the same as in the webshop so you can check your allowed region shipping destinations in My account > Saved addresses.

order_lines

- **order_line_id**
If an order has multiple lines then each line uses a unique ID, you can assign any number here. As many order lines as wanted can be added (up to a Max of 50 items per order). This number will be replaced by midocean order line numbers in our system.

- **sku (only for UNPRINTED & SAMPLE orders)**
The product specific variant SKU which is the {master code}-{variant code}, for textiles it's {master_code}-{variant_code}-{size_code}
"sku" field in product information 2.0
- **master_code (only for PRINTED orders)**
The product specific master {master_code}
"master_code" field in product information 2.0 or printing data 1.0
- **expected_price**
Can be set to 0 always if "check_price" is set to false since the calculation is done by our systems.
If "check_price" is set to true you have to specify the total price of the order line here.

Print Configuration Fields

- **id**
ID of the print position
"position_id" field in print data 1.0
- **print_size_height**
Area height in mm, must be lower than the maximum height size
"max_print_size_height" field in print data 1.0
- **print_size_width**
Area width in mm, must be lower than the maximum width size
"max_print_size_width" field in print data 1.0
- **printing_technique_id**
ID of the print technique
"position_id" field in print data 1.0
- **number_of_print_colours**
The number of colours that your logo has
"max_colours" fields in print data 1.0
NOTE: Techniques with "0" "max_colours" have no limit on number of colours
- **print_artwork_url**
URL in which the logotype to be printed can be found.
- **print_mockup_url**
URL in with the representation / indications of the logotype position in the print area
- **print_instruction**
Instructions for our pre-press department (Max. 300 characters)
- **color**
Pantone colours that the logotype will be printed with (Max. 12 characters)
Note that the number of "color" specified depends on the "number_of_print_colours"
- **item_color_number**
The code of the variant
"variant_color" in print data 1.0 OR "color_code" field in product information 2.0
- **item_size (ONLY FOR TEXTILE ITEMS)**
Specifies the textile size

Order Request format (JSON)

Regular Order

These are the usual order types, plain and unprinted, they can have multiple lines a

NOTE: It's not possible to order textile products with NORMAL orders, use SAMPLE or PRINT instead.

Here's an example of the format for a NORMAL order with one line:

```
{
  "order_header": {
    "check_price": "",
    "proof_approved": "",
    "timestamp": "",
    "currency": "",
    "po_number": "",
    "contact_name": "",
    "contact_email": "",
    "shipping_address": {
      "contact_name": "Example", * Mandatory Field
      "company_name": "",
      "street1": "Example", * Mandatory Field
      "postal_code": "0000",
      "city": "Example", * Mandatory Field
      "region": "Region",
      "country": "ES", * Mandatory Field
      "email": "contact@mail.com",
      "phone": "0000000000"
    },
    "order_type": "", * Mandatory Field Use PRINT, NORMAL or SAMPLE.",
    "preferred_shipping_date": "",
  },
  "order_lines": [
    {
      "order_line_id": "",
      "sku": "KC3314-32", * Mandatory Field
      "variant_id": "10144631", * Mandatory Field
      "quantity": "1", * Mandatory Field
      "expected_price": "0"
    }
  ]
}
```

Print Order

With print orders you can specify the printing configuration for each product print position for each order line and also order multiple variants (colors or sizes) in the same line. You can't combine printed and unprinted in the same order request.

NOTE: To make printed orders it's needed to integrate the Print Data and Print Pricelist APIs to get the details.

Here's an example of the format for a PRINT order with one line :

```
{
  "order_header": {
    "check_price": "",
    "proof_approved": "",
    "timestamp": "",
    "currency": "",
    "po_number": "",
    "contact_name": "",
    "contact_email": "",
    "shipping_address": {
      "contact_name": "Name", * Mandatory Field
      "company_name": "",
      "street1": "Address", * Mandatory Field
      "postal_code": "0000",
      "city": "City", * Mandatory Field
      "region": "Region",
      "country": "ES", * Mandatory Field
      "email": "contact@mail.com",
      "phone": "0000000000"
    },
    "order_type": "PRINT", * Mandatory Field

    "preferred_shipping_date": "2023-03-03"
  },
  "order_lines": [
    {
      "order_line_id": "10",
      "master_code": "AR1804", * Mandatory Field
      "quantity": "",
      "expected_price": "",
      "printing_positions": [
        {
          "id": "FRONT", * Mandatory Field
          "print_size_height": "24", * Mandatory Field
          "print_size_width": "45", * Mandatory Field
          "printing technique id": "S2", * Mandatory Field

```

```

        "number_of_print_colors": "1", * Mandatory Field
        "print_artwork_url": "https://www.url.com/logo", *
Mandatory Field
        "print_mockup_url": "https://www.url.com/mockup",
        "print_instruction": "None",
        "print_colors": [
            {
                "color": "PMS 0 C"
            }
        ]
    },
    "print_items": [
        {
            "item_color_number": "05", * Mandatory Field
            "quantity": "10" * Mandatory Field
        }
    ]
}
]
}

```

Sample Order

These type of orders and Normal orders are structurally almost the same, difference is that these orders are locked to 5 pieces or less for each line and that you can order unprinted textile with these.

They also can't have printing

NOTE: With sample orders you can order unprinted textile, but only up to 5 pieces.

In this example you can see a sample order with 1 line:

```

{
  "order_header": {
    "check_price": "",
    "proof_approved": "",
    "timestamp": "",
    "currency": "",
    "po_number": "",
    "contact_name": "",
    "contact_email": "",
    "shipping_address": {
      "contact_name": "Name", * Mandatory Field
      "company_name": "",
      "street1": "Address", * Mandatory Field
      "postal_code": "0000",

```

```
        "city": "City", * Mandatory Field
        "region": "Region",
        "country": "ES", * Mandatory Field
        "email": "contact@mail.com",
        "phone": "0000000000"
    },
    "order_type": "SAMPLE", * Mandatory Field

    "preferred_shipping_date": "2022-03-03" * Mandatory Field
},
"order_lines": [
    {
        "order_line_id": "",
        "sku": "KC3314-32", * Mandatory Field
        "variant_id": "10144631", * Mandatory Field
        "quantity": "1", * Mandatory Field
        "expected_price": "0"
    }
]
}
```

Order Response

After the POST request is made our systems will respond with the “**status_code**” and “**status_text**” to indicate if the order went through or failed. To check on possible error resolutions go [here](#).

Response for all types of orders is the same, only difference is for PRINT orders the print costs will show a the price, for the rest they will be at 0.

In this example you can see the response to the PRINT order that we set as example, at the header the prices are indicated. The rest of the response is just the order details we specified on the request.

```
{
  "order_header": {
    "status_code": "0",
    "status_text": "Sales order successfully created",
    "direct_confirmation": "",
    "proof_approved": "false",
    "check_price": "false",
    "currency": "EUR",
    "po_number": "test order 2.0",
    "order_type": "PRINT",
    "order_number": "2623021",
    "total_item_price": "68,00",
    "total_print_costs": "208,40",
    "total_net_price": "297,35",
    "freight_charge": "16.000000000 ",
    "tax": "68,39",
    "print_setup": "140,00",
    "print_cost": "67,40",
    "print_handling": "1,00",
    "shipping_address": {
      "contact_name": "Name",
      "company_name": "Company",
      "street1": "Address",
      "street2": "",
      "street3": "",
      "postal_code": "0000",
      "city": "City",
      "region": "Region",
      "country": "ES",
      "email": "contact@mail.com",
      "phone": "000000000"
    }
  },
}
```

Error Resolutions

When making an order through the API it is possible that the system exports an error with a message. Usually the responds with “**status_code**” **99** and the error itself is explained in the “**status_text**” field. The message usually is pretty self-explanatory but for some cases here, is a list.

Insufficient stock

The product set on the order request doesn't have enough stock, or the preferred shipping date is set too soon for the stock to arrive to the warehouse.

To check stock availability, you can import the [stock file](#)

Here's an example error 999 response:

```
{
  "order_header": {
    "status_code": "999",
    "status_text": "Insufficient stock",
    "direct_confirmation": "",
  },
}
```


Invalid date pattern or input

The “preferred_shipping_date” or “timestamp” field doesn’t follow the correct format, they should be set as in here:

- *Delivery date: YYYY-MM-DD | Example: 2023-03-20*
- *Timestamp: YYYY-MM-DD then a T then HH:MM:SS | Example: 2023-03-03T08:17:00*

An order with your reference already exist

When entering an API order if your po_number is the same as it was for a previous order the system will respond with an error to avoid duplicated orders.

You can bypass the need to set a different po_number for each order by leaving the field empty

Destination country is not allowed

Based on your API key region there’s a limitation on the shipping countries allowed, if the country code specified in the “country” field is not valid the system will respond with this error.

For more information on which countries are allowed for your sales organization please contact your regular midocean salesperson.

The list of countries is 1 to 1 the same as in the webshop so you can check your allowed region shipping destinations in My account > Saved addresses.

For more general errors with the API you can check [Appendix C.1](#)

Order Details 2.1

By using this API the Distributors of midocean will be able to retrieve all order information, details and updated tracking from any order they already placed in the system

The **Order details & tracking** Rest API has to be called at the following URL by making a GET call:

- https://api.midocean.com/gateway/order/2.1/detail?order_number=Order Number

**Change the “Order Number” part for the specific order number to check*

The OUTPUT will be a file with the updated order details from midocean systems, here’s an example response:

First part, the order_header gives details about the order, the customer and the shipping address. For instance the order creation date, the order status which can be “OPEN”, “COMPLETED” or “CLOSED” order found by the call and the customer that made it as well as about the shipping address, the creation date, the sales organization of the customer (see [Sales organizations](#) table), the order status, etc.

About order status, the order can be on

- **OPEN**
- **COMPLETED**
- **CLOSED**

```
"order_header": {  
  "order_found": "true",  
  "order_number": "2622891",  
  "order_date": "2023-08-10",  
  "order_status": "OPEN",  
  "sales_org": "0101",  
  "currency": "EUR",  
  "customer_number": "80865772",  
  "contact_person": "Customer Care",  
  "po_number": "test order 2.0",  
  "order_type": "PRINT",  
  "shipping_address": {  
    "company_name": "Company",  
    "contact_name": "Name",  
    "street1": "Address",  
    "street2": "",  
    "postal_code": "0000",  
    "city": "City",  
    "region": "",  
    "country": "ES",  
    "email": "contact@mail.com",  
    "phone": "000000000"
```

Order pricing & transport details – This next section shows the total: product, printing, discount, transport and tax prices. It also displays the transport mode (Standard or Express) and the incoterms.

```
"total_item_price": "39.24",  
"discounts": "0.0",  
"total_print_costs": "164.59",  
"freight_charge": "16.0",  
"total_net_price": "219.83",  
"tax": "50.56",  
"total_gross_price": "270.39",  
"incoterms": "DAP",  
"delivery_service": "STANDARD"
```

```

"order_lines": [
  {
    "order_line_id": "100",
    "master_code": "AR1804",
    "master_id": "40000190",
    "quantity": "1",
    "item_price": "0.0",
    "print_setup": "50.0",
    "print_cost": "5.76",
    "print_handling": "0.24",
    "proof_url": "",
    "shipping_status": "OPEN",
    "shipping_date": "2023-08-22",
    "print_items": [
      {
        "sku": "AR1804-03",
        "variant_id": "10168709",
        "quantity": "6",
        "item_price": "1.2"
      }
    ],
    "printing_positions": [
      {
        "id": "FRONT",
        "print_size_height": "190",
        "print_size_width": "120",
        "printing_technique_id": "B",
        "number_of_print_colors": "1"
      }
    ]
  }
]

```

Notes:

<Order_Status> values: Open/Completed/Cancelled
 <Order_Type> values: PRINT/NORMAL/SAMPLE
 <Proof_Status> values: InProgress/ArtworkRequired/WaitingApproval/Approved
 <Incoterms> values: Exworks/DAP
 <Service> values: Standard/Express
 <Delivery_Status> values: Open/Shipped

Approve Proof 1.0

By using this API the Distributors of midocean will be able to confirm an existing proof on an order. This API can only be used on proofs that have status "WaitingApproval" returned from the Order Detail API.

The **Approve Proof** Rest API has to be called at the following URL by making a POST call:

➤ <https://api.midocean.com/gateway/proof/1.0/approve>

The BODY required for the call is:

```

{
  "order_number": "xxxxxxx",
  "order_line_id": "100"
}

```

Where "order_number" and "order_line_id" specify the affected order/order line combination, numbers to be used are provided in the order detail API response.

Expected OUTPUT for successful calls is 200 status code and proof status changed to "Approved" in Order Detail API.

Reject Proof 1.0

By using this API the Distributors of midocean will be able to reject an existing proof on an order. This API can only be used on proofs that have status "WaitingApproval" returned from the Order Detail API.

The **Reject Proof** Rest API has to be called at the following URL by making a POST call:

➤ <https://api.midocean.com/gateway/proof/1.0/reject>

The BODY required for the call is:

```
{
  "order_number": "xxxxxxx",
  "order_line_id": "100",
  "rejection_code": 3,
  "rejection_comment": "Your rejection comments or instructions here",
  "additional_files": ["https://www.url.com/file1",
"https://www.url.com/file2", "..."],
  "new_artworks": ["https://www.url.com/artwork1",
"https://www.url.com/artwork2", "..."],
}
```

Where:

- "order_number" and "order_line_id" specify the affected order/order line combination, numbers to be used are provided in the order detail API response.
- "rejection_code" refers to the numeric code of the rejection reason to be used (see Appendix E for full table)
- "rejection_comment" is a placeholder for any additional instructions that you would like to give to prepress department
- "additional_files" and "new_artworks" are arrays that you can fill with different URLs for instructions (additional_files) or new artworks ("new_artworks" in combination with rejection reason "Change artwork").

Expected OUTPUT for successful calls is 200 status code and proof status changed to "InProgress" in Order Detail API.

Add Artwork 1.0

By using this API the Distributors of midocean will be able to submit an artwork to be used by prepress to prepare a new proof for an order. This API can only be used on proofs that have status "ArtworkRequired" returned from the Order Detail API.

The **Reject Proof** Rest API has to be called at the following URL by making a POST call:

➤ <https://api.midocean.com/gateway/proof/1.0/addartwork>

The BODY required for the call is:

```
{
  "order_number": "xxxxxxx",
  "order_line_id": "100",
  "position_name": "ROUNDScreen",
  "artwork_url": "https://www.url.com/artwork"
}
```

Where:

- “order_number” and “order_line_id” specify the affected order/order line combination, numbers to be used are provided in the order detail API response.
- “position_name” refers to the print position where the artwork will be used, provided in the order detail API response.
- “artwork_url” is a unique URL with the artwork to be used

Expected OUTPUT for successful calls is 200 status code and proof status changed to “InProgress” in Order Detail API.

APPENDIX A: Rest API Integration using Postman

Depending on the software you are using on your website or ERP system you need an XML parser and communication software to integrate with our API.

A common library used for communication is Curl which is available for Java, C/C++, PHP, etc. Website: <https://curl.haxx.se/libcurl/>

For testing the communication with our Rest API, you can have a look at **Postman**. Postman is collaboration platform for API development which is available in open source on Windows, Linux and Mac OS.

Website: <https://www.postman.com/>

When testing with our Rest API it is handy to have a test environment available.

Instead of using live URLs (located at <https://api.midocean.com>) you may use our test environment, reachable at <https://apitest.midocean.com>.

(NOTE: You have to request a separate key to use this environment)

Here's a quick guide of how to create and manage your Postman Rest API calls.

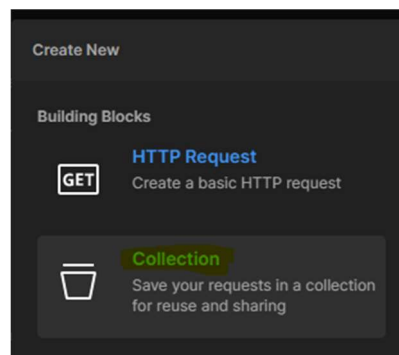
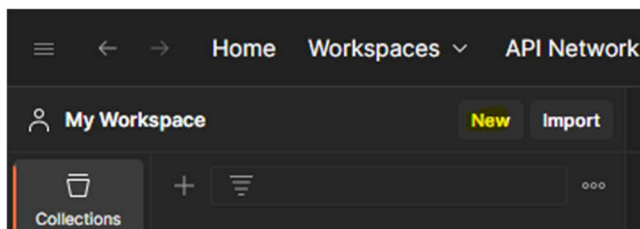
- Download Postman here: <https://www.postman.com/downloads/>

The Postman app

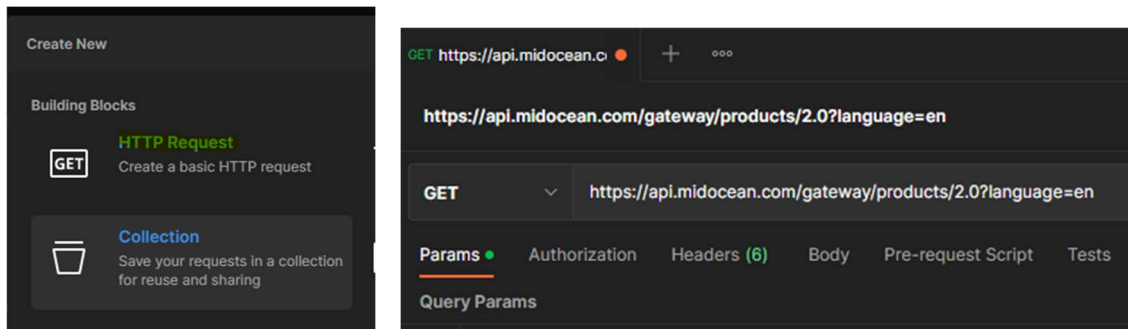
Download the app to get started with the Postman API Platform.

 Windows 64-bit

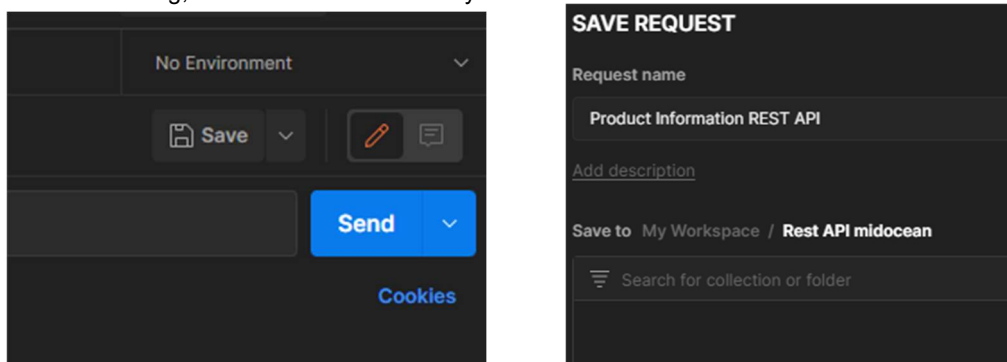
- After installing Postman, click on **New** button, then select **Collection**. A collection acts a folder to store every Rest API call.



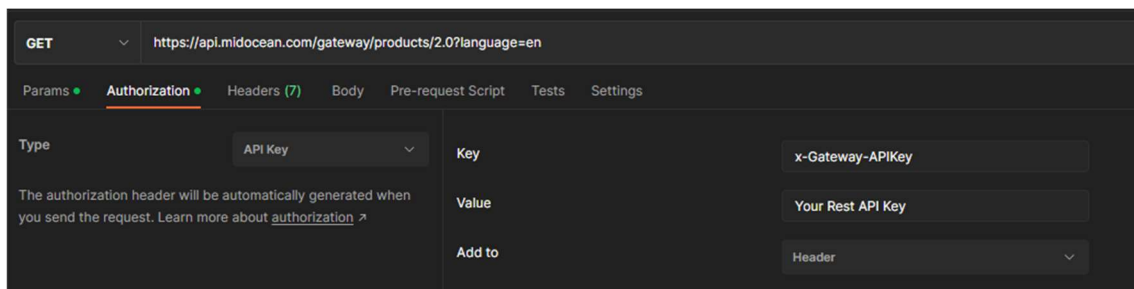
- To add the API calls follow same procedure, but select **HTTP Request**
- Once created you can change the name to know which API it is, in this example we create a request for Product Information file, but the procedure is the same for all files, just changing the link.
- In the request URL type the url of the specific request you are going to make, in this case: **<https://api.midocean.com/gateway/products/2.0?language=en>**



- Press on the **Save** button to save the HTTP Request, set a name to identify which call you're making, then select the collection you created and click **save** to add it in the collection folder



- Now last step is to add the authorization, meaning the Rest API key. To do that select the **Authorization** tab
- In Type, select **API Key**, then put the next values
 - Key: **x-Gateway-APIKey**
 - Value: **(your Rest API key)**



- If you want to choose between **.xml** and **.json** output; Go to **Headers** tab and set create a new header with
 - Key: **Accept**
 - Value:
 - **text/xml** - to request in xml
 - **text/json** - to request in json

Params		Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings
<input checked="" type="checkbox"/>	x-Gateway-APIKey					①	Your Rest API Key
<input checked="" type="checkbox"/>	Postman-Token					①	<calculated when request is sent>
<input checked="" type="checkbox"/>	Host					①	<calculated when request is sent>
<input checked="" type="checkbox"/>	User-Agent					①	PostmanRuntime/7.31.0
<input checked="" type="checkbox"/>	Accept					①	*/*
<input checked="" type="checkbox"/>	Accept-Encoding					①	gzip, deflate, br
<input checked="" type="checkbox"/>	Connection					①	keep-alive
<input checked="" type="checkbox"/>	Accept						text/xml
	Key						Value

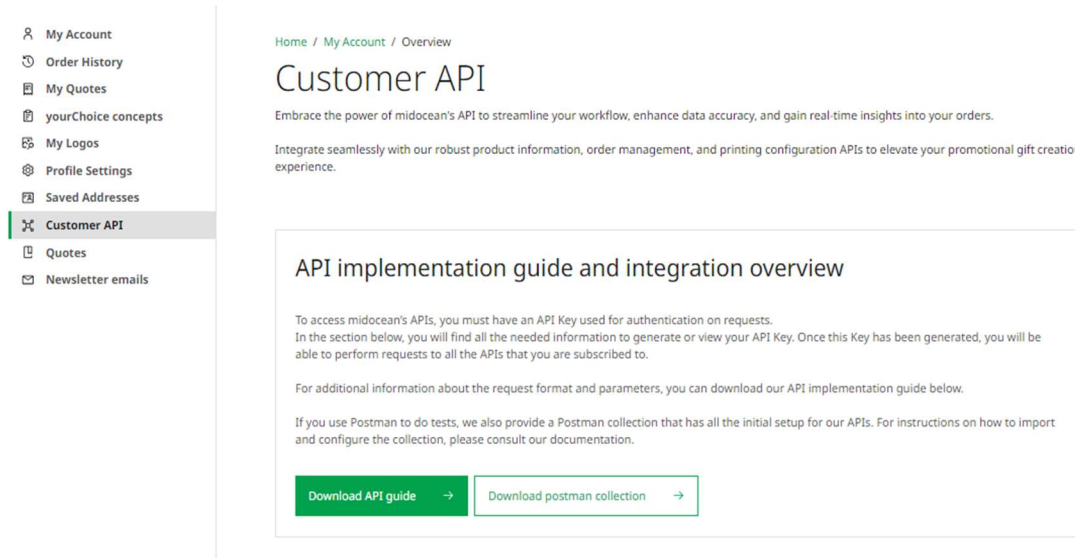
- Now save the call and click on **Send** to request the data, and that's it! Later on the guide we'll present you the many options for API information and service calls you can make.

APPENDIX B: How to Download and Import Postman Collection from Customer API Page JSON or URL

Postman Collection: JSON

To obtain the JSON Postman collection, customers need to access the Midocean website.

- Navigate to the Customer API section and click on 'Download Postman Collection'."

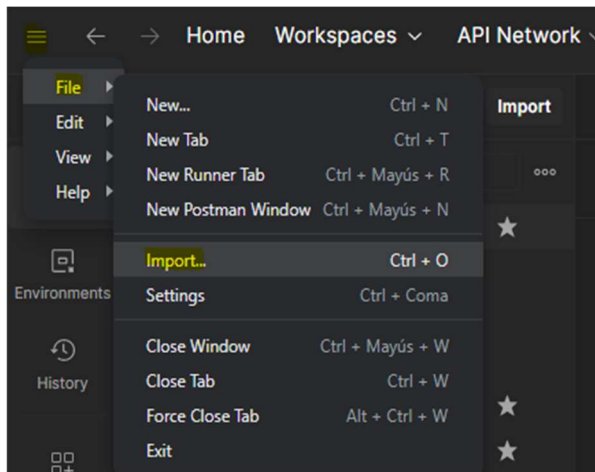


After successfully downloading the Postman Collection as a ZIP file from the Midocean website, proceed with the following:

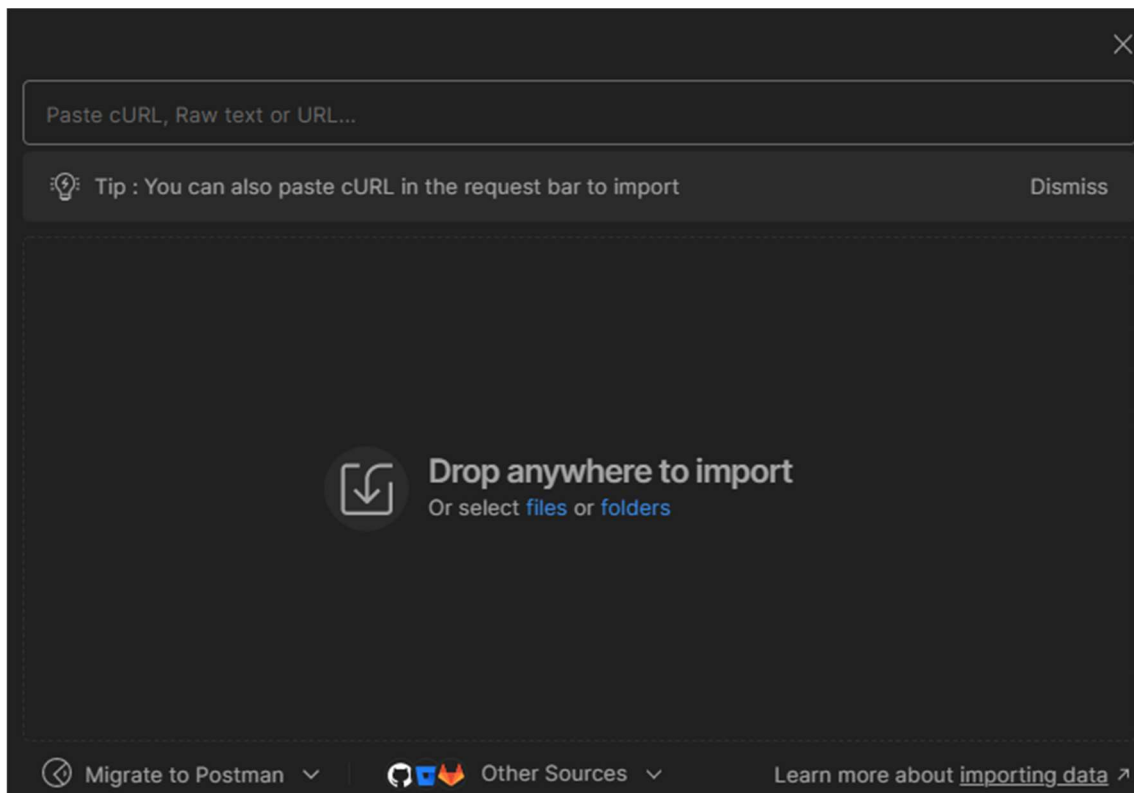
- Extract the contents of the ZIP file on your computer's file system.

To import the extracted JSON file into the Postman application, adhere to the subsequent steps:

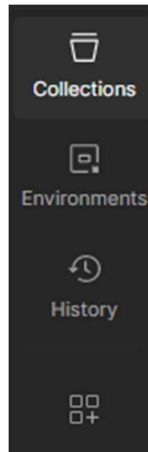
- Launch the Postman application on your computer.
- Access the "File" menu situated at the top-left corner of the Postman interface.
- From the dropdown menu, select the option labeled "Import".



- Locate the extracted midocean_postman_collection.json file on your computer.
- Drag and drop the JSON file into the designated import area within the Postman interface.

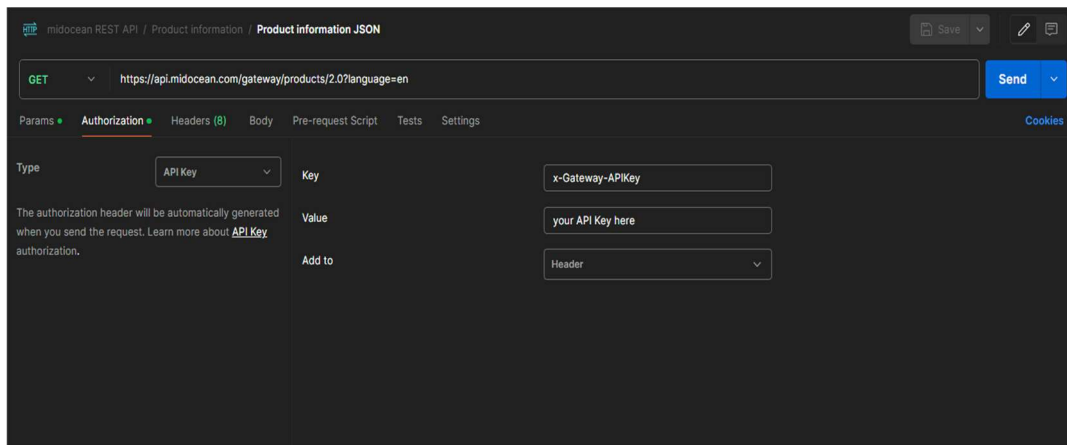


- Once successfully imported, the Postman collection is readily accessible within the Postman application:
- Navigate to the "Collections" tab situated in the sidebar of the Postman interface.
- Locate the imported collection within the list of available collections.



After configuring the collection within Postman, it's imperative to authenticate your requests by integrating your unique API Key:

- Navigate to the relevant endpoint within the imported collection.
- Locate the designated field labeled "Your API Key here".
- Replace the placeholder text with your personal API Key provided by Midocean.



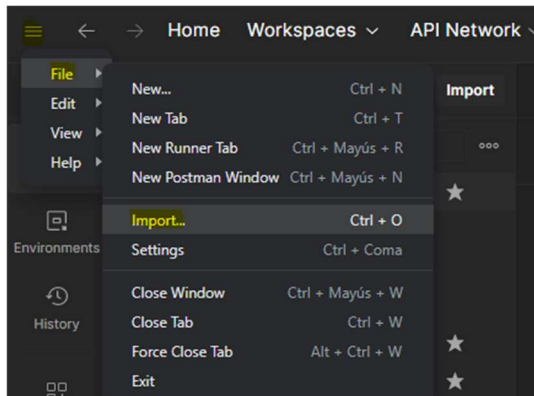
With your API Key successfully integrated, you're now equipped to initiate requests to the API endpoint:

- Ensure all necessary parameters are appropriately configured within the request.
- Execute the request by clicking the "Send" button.
- Observe the response status and data returned to verify successful communication with the Midocean API

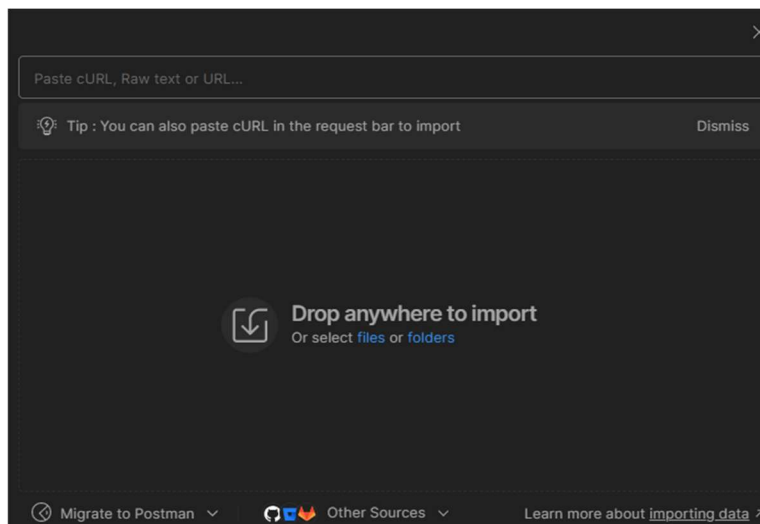
Postman Collection: URL

To obtain the JSON Postman collection URL, customers need to follow next steps.

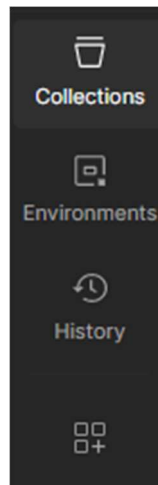
- Launch the Postman application on your computer.
- Access the "File" menu situated at the top-left corner of the Postman interface.
- From the dropdown menu, select the option labeled "Import".



- After opening the import window, copy the following URL and paste it into the Postman
 - <https://midocean.com/midoceanAPI>

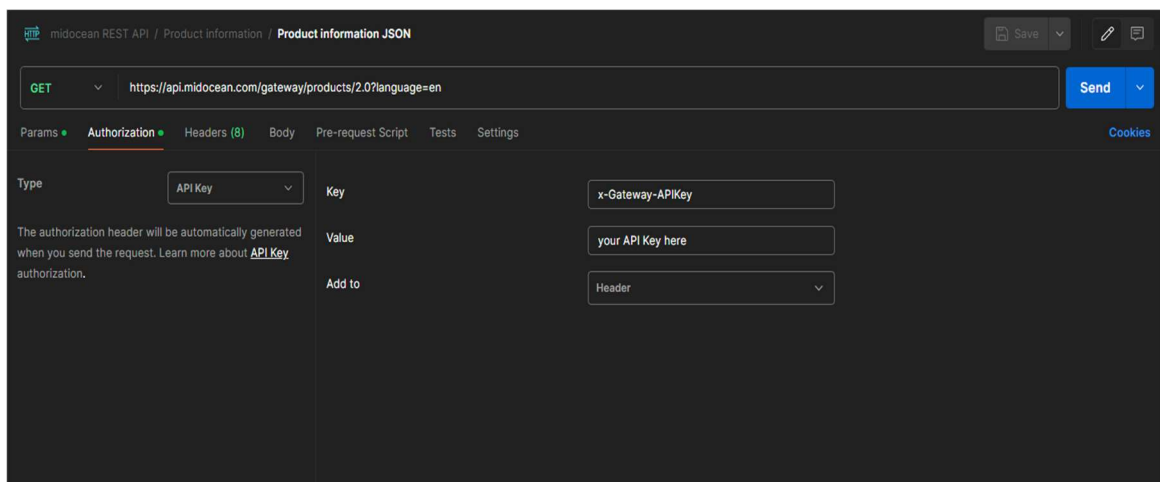


- Once successfully imported, the Postman collection is readily accessible within the Postman application:
- Navigate to the "Collections" tab situated in the sidebar of the Postman interface.
- Locate the imported collection within the list of available collections.



After configuring the collection within Postman, it's imperative to authenticate your requests by integrating your unique API Key:

- Navigate to the relevant endpoint within the imported collection.
- Locate the designated field labeled "Your API Key here".
- Replace the placeholder text with your personal API Key provided by midocean.



With your API Key successfully integrated, you're now equipped to initiate requests to the API endpoint:

- Ensure all necessary parameters are appropriately configured within the request.
- Execute the request by clicking the "Send" button.
- Observe the response status and data returned to verify successful communication with the midocean.

APPENDIX C: API Integration Errors

When using our API services there are an assortment of errors, sometimes they happen because of connection issues, error with the setup for an API call a wrong parameter, wrong link or ID and many more variables.

Here is a compilation of errors and the possible workarounds, starting with:

Error “503 Service Temporarily Unavailable”

This error means that communication to the API service is not available or that there are connection issues, this can be for many reasons:

- If connecting to the TEST Environment API.
 - Note that the TEST Environment is switched off over the evening, during out of business hours in Europe.

Error “Unauthorized Application Request”

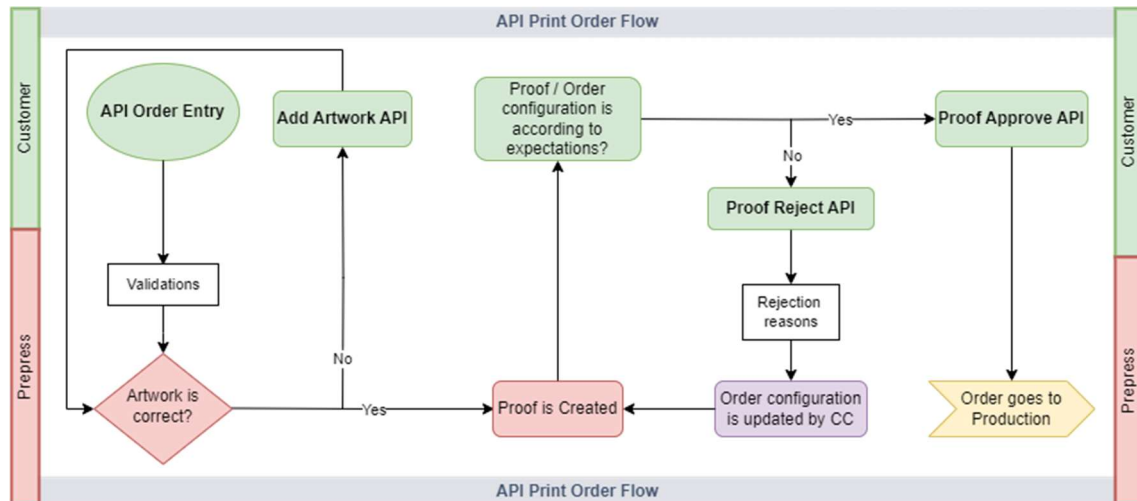
This error means that the authorization to make the request is not working, can happen for the following reasons:

- The Rest API Key is not setup correctly.
 - **Solution:** Make sure the Rest API Key is written correctly with the right header “**Value**” and “**Key**”
If unsure you can test the API calls using Postman
- The Rest API Key is invalid or doesn't have the right API permissions.
 - **Solution:** From the Webshop, go to Account > Customer API and click on “**Request new API key**” button to request an API key.
Select the missing APIs and click on “**Subscribe**” to gain permission to access the calls.
- Make sure you're using the right API key for the right environment, Test and Production use different keys.

Error “404 Not found” or “403 Forbidden”

Both errors usually mean the endpoint URL is not correct | List of API endpoint URLs found [here](#).

APPENDIX D: Printed API Order Flow



APPENDIX E: List of rejection reasons

Code	Description
1	This is a repeat order
2	Change the print technique
3	The logo looks blurred
4	The proof is not according to my expectations
6	Change the imprint size
8	Change the logo
9	Move logo or artwork within selected position
10	Change the print position
11	Change the number of imprint colours
12	Change the item colour
13	Change the imprint colour
14	Remove or change part of the logo
15	Change the item quantity
16	I'd like to approve a previous proof version