

BEST BUY MARKETPLACE™ SELLER API SPECIFICATION



This document is confidential and proprietary to Best Buy Canada Ltd. and may not be reproduced or disclosed to anyone without the prior written permission of Best Buy Canada Ltd., 8800 Glenlyon Parkway, Burnaby, BC V5J 5K3.

Table of Contents

Best Buy Marketplace Seller REST API - General notes.....	5
Account/Offers/Orders/Shipping.....	10
AF01 - Get the list of any additional fields	10
IV01 - List invoices.....	12
IV02 - Download an invoice.....	14
OF01 - Import offers: import file to add offers (and products).....	15
OF02 - Get import information and stats.....	17
OF03 - Get error report file for an import.....	20
OF21 - List offers for a shop	21
OF22 - Get information of an offer	27
OF24 - Create, update or delete offers.....	32
OF61 - Get the list of offer states	36
OR11 - List orders	38
OR21 - Accept or refuse order lines of an order which are in "WAITING_ACCEPTANCE" state	53
OR23 - Update carrier tracking information of a given order.....	55
OR24 - Update the shipment of the order which is in "SHIPPING" state to "SHIPPED" state	57
OR28 - Demand refunds on order lines.....	58
OR72 - Lists order's documents.....	63
OR73 - Download one or multiple documents associated to one or multiple orders	65
OR74 - Upload documents to associate to an order	67
P11 - List offers for each product given.....	70
P31 - Get products for a list of product's references.....	77
RE02 - List of reasons by type.....	79
S30 - Lists shop's documents	81
S31 - Downloads one or multiple shop documents	83
SH21 - List all carriers.....	84
Catalog/Products.....	86
H11 - List hierarchies related (parents and children) to the given hierarchy	86
P41 - Import offers: import file to add products	88
P42 - Get import status.....	90
P44 - Get errors report file for an import	92
P45 - Get products report file for an import	93
P46 - Get transformed file for an import	94
P47 - Get transformation errors report file for an import.....	95
PM11 - Get attributes configuration	96
PM31 - Returns values for given operator's values list	98
Appendecies	100
Appendix A - Retrieve your shop API key	100
Appendix B - Download form templates for OF01 and P41.....	101
Appendix C - Postman Compatible JSON Collection.....	103

Changelog:

Version	Change notes
January 2016	Initial
March 2016	Added Appendecies A, B, and C Updated OF01 and P41 Removed support for OR74 and S32

Best Buy Marketplace Seller REST API - General notes

Authentication

You can authenticate through API by sending your API key in a header:

```
curl -H "Authorization: 123456" https://[HOSTNAME]/api/[API-URL]
```

See [Appendix A](#) for instruction on how to retrieve your API key..

Request Content-Type

You can choose between json or xml format.

Use Content-Type Header to set the request content-type.

```
curl -H "Content-Type: application/json" https://[HOSTNAME]/api/[API-URL]
```

Accepted values are

- application/json
- application/xml
- multipart/form-data (*Allow to send file and form data. See the [cURL samples](#) at the end of this page*)

Response Content-Type

For no-binary response you can choose between json or xml format.

Use Accept Header to choose response content-type.

```
curl -H "Accept: application/json" https://[HOSTNAME]/api/[API-URL]
```

Accepted values are:

- application/json
- application/xml
- text/csv

SSL only

All request should use SSL protocol.

UTF-8 encoding

All text data is encoded in UTF-8. The binary content is not concerned.

Date format

All timestamps are returned in ISO 8601 format:

yyyy-mm-dd'T'HH:mm:ss'Z'

ISO 8601 considers the time zone.

Note that you can use partial timestamps, but keep in mind that in this case, timestamps are expected to be in UTC timezone

The examples below are equivalent:

2012-08-29T04:01:00Z

2012-08-29T04:09:00+08:00

2012-08-29T04:01:00

2012-08-29T04:01

Please note that query parameters must be URL encoded, i.e. when used into a query parameter, character '+' must be encoded as '%2B'. For instance, to send a date in a query parameter, the following syntax should be used:

2012-08-29T04:09:00%2B08:00

API Return Codes

Best Buy Marketplace API uses standard HTTP codes.

Success Codes

200

OK - After a successful GET.

Returns the resources

201

Created - After a successful POST.

Returns The URI of the new resource. This URI is in the response header.

204

No Content - After a successful PUT or DELETE.

Does not return anything.

Error Codes

400

Bad Request - Parameter errors or bad method usage.

Bad usage of the resource. Example: a required parameter is missing, a parameter with bad format, query on data which are not in expected state.

Response example

```
{"status":400,"message":"Parameter 'offer' is required"}
```

401

Unauthorized - Call of API without authentication

Add authentication information or use a valid authentication token.

Response example

```
{"status":401,"message":"Authentication Failed: Request does not have authentication token"}
```

403

Forbidden - Access to the resource is denied

Current user cannot access to the resource.

Response example

```
{"status":403,"message":"Access Denied"}
```

404

Not Found - The resource does not exist

The resource URI does not exist or the asking resource does not exists (for current user).

Response example

```
{"status":404,"message":"Not Found"}
```

405

Method Not Allowed - The http method (GET, POST, PUT, DELETE) is not allowed for this resource

See documentation to know accepted method.

Response example

```
{"status":405,"message":"Method Not Allowed"}
```

406

Not Acceptable - Asked response content-type is not available for this resource.

See documentation to use correct Accept Header for this request.

Response example

```
{"status":406,"message":"Not Acceptable"}
```

415

Unsupported Media Type - The entity content-type sent to server is not supported.

See documentation to use correct Content-type to send data.

Response example

```
{"status":415,"message":"Unsupported Media Type"}
```

500

Internal Server Error - The server encountered an unexpected error.

Response example

```
{"status":500,"message":"Internal Server Error"}
```

Pagination and sort

Paginate the results

Some resources support pagination. In this case, you can use max and offset parameters:

max

The parameter max is used to indicate the maximum number of items returned by page. This parameter is optional. The default value of this parameter is 10. The maximum value of this parameter is 100.

Example: get only the first 5 offers

```
_ /api/offers?max=5
```

offset

The parameter offset is used to indicate the index of the item of the results corresponding to the first item in the page returned. This parameter is optional. The default value of this parameter is 0, which means that no offset is applied.

Example: get offers only from the 5th offer

```
_ /api/offers?offset=4
```

With pagination, the URL of the previous and/or next page can be returned in the header's attribute Link of the response.

Example: get only 5 offers from the 6th offer

Request

```
_ /api/offers?max=5&offset=5
```

Header of the response

```
Link: ; rel="previous", ; rel="next"
```

Sort the results

Some resources support to be sorted. In this case, you can use sort and order parameters:

```
sort
```

The parameter sort is used to indicate how the results should be sorted. This parameter is optional. The possible values for this parameter are defined in resources. The default value of this parameter is defined in resources.

Example: get offers sorted by price

```
_ /api/offers?sort=price
```

```
order
```

The parameter order is used to indicate the sort order. This parameter is optional. The possible values for this parameter are asc (default) or desc.

Example: get offers sorted by price in descending order

```
_ /api/offers?sort=price&order=desc
```

cURL Samples

Here is the cURL parameters you need to use when testing the API with it.

GET

```
curl -X GET https://[HOSTNAME]/api/[API-URL]
```

PUT (JSON or XML content type)

```
curl -X PUT -H "Content-Type: application/json" -H "Accept: application/json" -d @data.json https://[HOSTNAME]/api/[API-URL]
```

PUT (CSV content type)

```
curl -X PUT -H "Content-Type: multipart/form-data" -H "Accept: application/json" -F "file=@data.csv;type=text/csv" https://[HOSTNAME]/api/[API-URL]
```

POST (JSON or XML content type)

```
curl -X POST -H "Content-Type: application/json" -H "Accept: application/json" -d @data.json https://[HOSTNAME]/api/[API-URL]
```

POST (CSV content type)

```
curl -X POST -H "Content-Type: multipart/form-data" -H "Accept: application/json" -F "file=@data.csv;type=text/csv" https://[HOSTNAME]/api/[API-URL]
```

Security

API output may contain HTML and javascript, before displaying data on your website make sure to escape it.

Account/Offers/Orders/Shipping

AF01 - Get the list of any additional fields

Method: GET

URL: /api/additional_fields

Recommended: 1/day

Maximum: 1/day

Query Parameters:

entities - The object on which the additional field applies. Ex: OFFER, SHOP

Output:

AdditionalFieldsBean - Representation of additional fields

Additional fields list

JSON:

```
{  
    "additional_fields": [ {  
        "accepted_values": [String],  
        "code": String,  
        "entity": "SHOP" | "OFFER" | "ORDER_LINE",  
        "label": String,  
        "required": Boolean,  
        "type": "STRING" | "DATE" | "NUMERIC" | "BOOLEAN" | "LINK" | "LIST"  
    } ]  
}
```

XML:

```
<additional_fields>  
    <additional_fields>  
        <additional_field> [zero or N]  
            <accepted_values>xsd:string</accepted_values> [zero or N]  
            <code>xsd:string</code>  
            <entity>"SHOP" | "OFFER" | "ORDER_LINE"</entity>  
            <label>xsd:string</label>  
            <required>xsd:boolean</required>  
            <type>"STRING" | "DATE" | "NUMERIC" | "BOOLEAN" | "LINK" |  
                "LIST"</type>  
        </additional_field>  
    </additional_fields>
```

```
</additional_fields>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
additional_fields	additional_field [zero or N]	true	false	List of additional fields representations
accepted_values	xsd:string [zero or N]	false	false	The list of accepted values for this additional field. This field is only present for additional fields of type 'LIST'.
code	xsd:string	true	false	Code of the additional field
entity	"SHOP" "OFFER" "ORDER_LINE"	false	false	The object on which the additional field applies
label	xsd:string	true	false	Label of the additional field
required	xsd:boolean	true	false	Whether or not this additional field will be required
type	"STRING" "DATE" "NUMERIC" "BOOLEAN" "LINK" "LIST"	true	false	Type of the additional field. <ul style="list-style-type: none"> • STRING chain of characters • DATE date in yyyy-mm-dd format • NUMERIC decimal value (eg 52.63) • BOOLEAN true or false • LINK valid URL • LIST list of values. Valid values are specified in accepted_values field

HTTP Return Codes:

200 – Success

HTTP Example: GET /api/additional_fields?entities=...

IV01 - List invoices

METHOD: GET

URL STRUCTURE: /api/invoices

Recommended: On each invoice list page view

Maximum: On each invoice list page view.

Note: this resource supports pagination (see documentation)

QUERY PARAMETERS:

start_date - [optional] Creation date for filtering (format : "yyyy-MM-dd'T'HH:mm:ss")

end_date - [optional] Creation date for filtering (format : "yyyy-MM-dd'T'HH:mm:ss")

SORT:

dateCreated - (default) Sort by creation date

OUTPUT:

InvoicesBean - List of invoices

List of invoices with pagination

JSON:

```
{  
  "invoices": [  
    {"currency_iso_code": "CAD",  
     "date_created": Date,  
     "end_time": Date,  
     "invoice_id": Number,  
     "start_time": Date,  
     "total_charged_amount": Number  
   ],  
   "total_count": Number  
}
```

XML:

```
<body>  
  <invoices>  
    <invoice> [zero or N]  
      <currency_iso_code>"CAD"</currency_iso_code>  
      <date_created>xsd:datetime</date_created>  
      <end_time>xsd:datetime</end_time>  
      <invoice_id>xsd:long</invoice_id>  
      <start_time>xsd:datetime</start_time>
```

```

<total_charged_amount>xsd:decimal</total_charged_amount>
</invoice>
</invoices>
<total_count>xsd:int</total_count>
</body>

```

Attribute details:

Elements				
Name	type	Required	Nillable	Description
invoices	invoice [zero or N]	true	false	List of invoices
currency_iso_code	"CAD"	true	false	The currency of the shop (iso format)
date_created	xsd:datetime	true	false	Creation date of the invoice
end_time	xsd:datetime	true	false	Ending date of the period
invoice_id	xsd:long	true	false	Identifier of the invoice
start_time	xsd:datetime	true	false	Starting date of the period
total_charged_amount	xsd:decimal	true	false	Total amount charged to the shop (including taxes)
total_count	xsd:int	true	false	Total count of invoices

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/invoices?start_date=...&end_date=...

IV02 - Download an invoice

METHOD: GET**URL STRUCTURE:** /api/invoices/{invoice}

invoice - [required] The identifier of the invoice

Recommended: On each invoice download**Maximum:** On each invoice download.**QUERY PARAMETERS:** N/A**OUTPUT:**

Binary - The PDF of the invoice

HTTP RETURN CODES:

200 - Success

404 - Invoice not found

HTTP Example: GET /api/invoices/{invoice}

OF01 - Import offers: import file to add offers (and products)**METHOD:** POST**URL STRUCTURE:** /api/offers/imports

File should be uploaded.

Returns the import identifier to track the status of the import.

Recommended: 12/hour**Maximum:** 60/hour**QUERY PARAMETERS:** N/A**FORM PARAMETERS:****file** - [required] Import file to upload. Use multipart/form-data with name 'file'.See [Appendix B](#) for csv format.**import_mode** - [required] [NORMAL, PARTIAL_UPDATE, REPLACE]**with_products** - [optional] This file also contains product information. Default: false**OUTPUT:**

OfferImportTrackingBean - Identifier of the import

Offer import tracking information

JSON:

{
 "import_id": Number,
 "product_import_id": Number
}

XML:

<offer_import_tracking>
 <import_id>xsd:long</import_id>
 <product_import_id>xsd:long</product_import_id>
</offer_import_tracking>

Attribute details:

Elements

Name	Type	Required	Nillable	Description
import_id	xsd:long	true	false	Identifier of the offer import
product_import_id	xsd:long	false	false	Identifier of the product import

PRODUCES:

application/json

application/xml

CONSUMES:

multipart/form-data

HTTP RETURN CODES:

201 - Created

HTTP RESPONSE HEADERS:

Location - URI of the new import.

HTTP Example:

POST /api/offers/imports

file=...&import_mode=...&with_products=...

OF02 - Get import information and stats

METHOD: GET

URL STRUCTURE: /api/offers/imports/{import}

import - [required] The identifier of the import

Information returned:

- Number of offers inserted
- Number of offers updated
- Number of import lines read
- Number of import lines in success
- Number of import lines in error

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS: N/A

OUTPUT:

OfferImportBean - Import information

Representation of an import

JSON:

```
{  
    "date_created": Date,  
    "has_error_report": Boolean,  
    "import_id": Number,  
    "lines_in_error": Number,  
    "lines_in_success": Number,  
    "lines_read": Number,  
    "mode": "NORMAL" | "PARTIAL_UPDATE" | "REPLACE",  
    "offer_deleted": Number,  
    "offer_inserted": Number,  
    "offer_updated": Number,  
    "reason_status": String,  
    "status": "WAITING_SYNCHRONIZATION_PRODUCT" | "WAITING" | "RUNNING" |  
    "COMPLETE" | "FAILED",  
}
```

XML:

```
<import>  
    <date_created>xsd:datetime</date_created>  
    <has_error_report>xsd:boolean</has_error_report>  
    <import_id>xsd:int</import_id>  
    <lines_in_error>xsd:int</lines_in_error>
```

```

<lines_in_success>xsd:int</lines_in_success>
<lines_read>xsd:int</lines_read>
<mode>"NORMAL" | "PARTIAL_UPDATE" | "REPLACE"</mode>
<offer_deleted>xsd:int</offer_deleted>
<offer_inserted>xsd:int</offer_inserted>
<offer_updated>xsd:int</offer_updated>
<reason_status>xsd:string</reason_status>
<status>"WAITING_SYNCHRONIZATION_PRODUCT" | "WAITING" | "RUNNING" |
"COMPLETE" | "FAILED"</status>
</import>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
date_created	xsd:datetime	true	false	Creation date of the import
has_error_report	xsd:boolean	true	false	Indicate if an error report exists
import_id	xsd:int	true	false	Identifier of the import
lines_in_error	xsd:int	true	false	Total count of lines in error
lines_in_success	xsd:int	true	false	Total count of lines in success
lines_read	xsd:int	true	false	Total count of lines read
mode	"NORMAL" "PARTIAL_UPDATE" "REPLACE"	true	false	Mode of the import
offer_deleted	xsd:int	true	false	Total count of deleted offers
offer_inserted	xsd:int	true	false	Total count of inserted offers
offer_updated	xsd:int	true	false	Total count of updated offers
reason_status	xsd:string	true	false	Status reason of the import
status	"WAITING_SYNCHRONIZATION_PRODUCT" "WAITING" "RUNNING" "COMPLETE" "FAILED"	true	false	Status of the import

PRODUCES:

application/json
application/xml

HTTP RETURN CODES:

200 – Success

HTTP Example: GET /api/offers/imports/{import}

OF03 - Get error report file for an import

METHOD: GET**URL STRUCTURE:** /api/offers/imports/{import}/error_report

import - [required] The identifier of the import

Recommended: 12/hour**Maximum:** 60/hour.**QUERY PARAMETERS:** N/A**OUTPUT:**

Binary - The error report file

HTTP RETURN CODES:

200 - Success

HTTP Example: GET /api/offers/imports/{import}/error_report

OF21 - List offers for a shop

METHOD: GET

URL STRUCTURE: /api/offers

Note: this resource supports pagination (see documentation)

Recommended: On each shop's offers page view

Maximum: On each shop's offers page view

QUERY PARAMETERS:

offer_state_codes - [optional] List of offer state codes.

sku - [optional] Offer's sku.

product_id - [optional] Product's sku.

favorite - [optional] filter only the favorite offers. (default false)

SORT:

totalPrice - (default) Sort by total price

price - Sort by price, total price

productTitle - Sort by product title, total price

OUTPUT:

ShopOffersDto - List of offers

List of offers of a shop with pagination

JSON:

```
{  
  "offers": [  
    {  
      "active": Boolean,  
      "available_end_date": Date,  
      "available_start_date": Date,  
      "category_code": String,  
      "category_label": String,  
      "channels": [String],  
      "currency_iso_code": "CAD",  
      "description": String,  
      "discount": {  
        "discount_price": Number,  
        "end_date": Date,  
        "origin_price": Number,  
        "ranges": [  
          {  
            "price": Number,  
            "quantity_threshold": Number  
          }]  
      },  
      "start_date": Date  
    }]
```

```
},
"favorite_rank": Number,
"logistic_class": {
  "code": String,
  "label": String
},
"min_quantity_alert": Number,
"min_shipping_price": Number,
"min_shipping_price_additional": Number,
"min_shipping_type": String,
"min_shipping_zone": String,
"offer_additional_fields": [
  {
    "code": String,
    "value": String
  }
],
"offer_id": Number,
"price": Number,
"price_additional_info": String,
"product_references": [
  {
    "reference": String,
    "reference_type": String
  }
],
"product_sku": String,
"product_title": String,
"quantity": Number,
"shop_sku": String,
"state_code": String,
"total_price": Number
],
"total_count": Number
}
```

XML:

```
<body>
<offers>
  <offer> [zero or N]
    <active>xsd:boolean</active>
    <available_end_date>xsd:datetime</available_end_date>
    <available_start_date>xsd:datetime</available_start_date>
    <category_code>xsd:string</category_code>
    <category_label>xsd:string</category_label>
    <channels>
      <channel_code>xsd:string</channel_code> [zero or N]
    </channels>
    <currency_iso_code>"CAD"</currency_iso_code>
    <description>xsd:string</description>
    <discount>
      <discount_price>xsd:decimal</discount_price>
      <end_date>xsd:datetime</end_date>
```

```

<origin_price>xsd:decimal</origin_price>
<ranges>
  <range> [zero or N]
    <price>xsd:decimal</price>
    <quantity_threshold>xsd:int</quantity_threshold>
  </range>
</ranges>
<start_date>xsd:datetime</start_date>
</discount>
<favorite_rank>xsd:int</favorite_rank>
<logistic_class>
  <code>xsd:string</code>
  <label>xsd:string</label>
</logistic_class>
<min_quantity_alert>xsd:int</min_quantity_alert>
<min_shipping_price>xsd:decimal</min_shipping_price>
<min_shipping_price_additional>xsd:decimal</min_shipping_price_additional>
<min_shipping_type>xsd:string</min_shipping_type>
<min_shipping_zone>xsd:string</min_shipping_zone>
<offer_additional_fields>
  <offer_additional_field> [zero or N]
    <code>xsd:string</code>
    <value>xsd:string</value>
  </offer_additional_field>
</offer_additional_fields>
<offer_id>xsd:long</offer_id>
<price>xsd:decimal</price>
<price_additional_info>xsd:string</price_additional_info>
<product_references>
  <product_reference> [zero or N]
    <reference>xsd:string</reference>
    <reference_type>xsd:string</reference_type>
  </product_reference>
</product_references>
<product_sku>xsd:string</product_sku>
<product_title>xsd:string</product_title>
<quantity>xsd:int</quantity>
<shop_sku>xsd:string</shop_sku>
<state_code>xsd:string</state_code>
<total_price>xsd:decimal</total_price>
</offer>
</offers>
<total_count>xsd:int</total_count>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
offers	offer [zero or N]	true	false	List of offers of a shop
active	xsd:boolean	true	false	
available_end_date	xsd:datetime	false	false	Available end date
available_start_date	xsd:datetime	false	false	Available start date
category_code	xsd:string	true	false	Code of the category product
category_label	xsd:string	true	false	Label of the category product
channels	xsd:string [zero or N]	true	false	List of the channel codes associated to the offer shop
currency_iso_code	"CAD"	true	false	The currency of the shop (iso format)
description	xsd:string	true	true	Description of the offer
discount	discount	false	true	Discount of the offer
discount_price	xsd:decimal	false	true	Discount price. Not to be read if discount with ranges is specified (i.e. discount with at least one range with quantity threshold greater than one).
end_date	xsd:datetime	false	true	Discount end date
origin_price	xsd:decimal	false	true	Origin price
ranges	range [zero or N]	false	false	Discount ranges
price	xsd:decimal	true	false	Price of the offer if the quantity threshold is reached
quantity_threshold	xsd:int	true	false	Minimum quantity to order for the given price to be applied
start_date	xsd:datetime	false	true	Discount start date
favorite_rank	xsd:int	false	false	Favorite rank if the offer is favorite and if the favorite offer feature is enabled
logistic_class	logistic_class	true	true	The logistic class associated to the offer
code	xsd:string	true	false	The code of the logistic class

label	xsd:string	true	false	The label of the logistic class
min_quantity_alert	xsd:int	true	false	
min_shipping_price	xsd:decimal	true	false	Unit price of the min shipping rate
min_shipping_price_additional	xsd:decimal	true	false	Additional unit price associated to the min shipping rate
min_shipping_type	xsd:string	true	false	Code of the shipping type associated to the min shipping rate
min_shipping_zone	xsd:string	true	false	Code of the shipping zone associated to the min shipping rate
offer_additional_fields	offer_additional_field[zero or N]	false	false	List of additional fields
code	xsd:string	true	false	Code of the additional field
value	xsd:string	true	false	Value of the additional field
offer_id	xsd:long	true	false	Identifier of the offer
price	xsd:decimal	true	false	Price of the offer
price_additional_info	xsd:string	false	true	Price additional information of the offer
product_references	product_reference[zero or N]	true	false	List of product's references
reference	xsd:string	true	false	Value of the reference
reference_type	xsd:string	true	false	Type of the reference
product_sku	xsd:string	true	false	Sku of the product
product_title	xsd:string	true	false	Title of the product
quantity	xsd:int	true	false	Available quantity of the offer
shop_sku	xsd:string	true	false	Shop's sku of the offer
state_code	xsd:string	true	false	State of the offer
total_price	xsd:decimal	true	false	Total price of the offer (price + min shipping rate)
total_count	xsd:int	true	false	Total count of offers

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET

/api/offers?offer_state_codes=...&sku=...&product_id=...&favorite=...

OF22 - Get information of an offer

METHOD: GET

URL STRUCTURE: /api/offers/{offer}

offer - [required] The identifier of the offer

Recommended: On each offer page view

Maximum: On each offer page view

QUERY PARAMETERS: N/A

OUTPUT:

ShopOfferDto - Representation of an offer

JSON:

```
        "value": String
    }],
    "offer_id": Number,
    "price": Number,
    "price_additional_info": String,
    "product_references": [
        {
            "reference": String,
            "reference_type": String
        },
        "product_sku": String,
        "product_title": String,
        "quantity": Number,
        "shop_sku": String,
        "state_code": String,
        "total_price": Number
    ]
}
```

XML

```
<body>
    <active>xsd:boolean</active>
    <available_end_date>xsd:datetime</available_end_date>
    <available_start_date>xsd:datetime</available_start_date>
    <category_code>xsd:string</category_code>
    <category_label>xsd:string</category_label>
    <channels>
        <channel_code>xsd:string</channel_code> [zero or N]
    </channels>
    <currency_iso_code>"CAD"</currency_iso_code>
    <description>xsd:string</description>
    <discount>
        <discount_price>xsd:decimal</discount_price>
        <end_date>xsd:datetime</end_date>
        <origin_price>xsd:decimal</origin_price>
        <ranges>
            <range> [zero or N]
                <price>xsd:decimal</price>
                <quantity_threshold>xsd:int</quantity_threshold>
            </range>
        </ranges>
        <start_date>xsd:datetime</start_date>
    </discount>
    <favorite_rank>xsd:int</favorite_rank>
    <logistic_class>
        <code>xsd:string</code>
        <label>xsd:string</label>
    </logistic_class>
    <min_quantity_alert>xsd:int</min_quantity_alert>
    <min_shipping_price>xsd:decimal</min_shipping_price>
    <min_shipping_price_additional>xsd:decimal</min_shipping_price_additional>

```

```

<min_shipping_type>xsd:string</min_shipping_type>
<min_shipping_zone>xsd:string</min_shipping_zone>
<offer_additional_fields>
  <offer_additional_field> [zero or N]
    <code>xsd:string</code>
    <value>xsd:string</value>
  </offer_additional_field>
</offer_additional_fields>
<offer_id>xsd:long</offer_id>
<price>xsd:decimal</price>
<price_additional_info>xsd:string</price_additional_info>
<product_references>
  <product_reference> [zero or N]
    <reference>xsd:string</reference>
    <reference_type>xsd:string</reference_type>
  </product_reference>
</product_references>
<product_sku>xsd:string</product_sku>
<product_title>xsd:string</product_title>
<quantity>xsd:int</quantity>
<shop_sku>xsd:string</shop_sku>
<state_code>xsd:string</state_code>
<total_price>xsd:decimal</total_price>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
active	xsd:boolean	true	false	
available_end_date	xsd:datetime	false	false	Available end date
available_start_date	xsd:datetime	false	false	Available start date
category_code	xsd:string	true	false	Code of the category product
category_label	xsd:string	true	false	Label of the category product
channels	xsd:string [zero or N]	true	false	List of the channel codes associated to the offer shop
currency_iso_code	"CAD"	true	false	The currency of the shop (iso format)
description	xsd:string	true	true	Description of the offer
discount	discount	false	true	Discount of the offer

discount_price	xsd:decimal	false	true	Discount price. Not to be read if discount with ranges is specified (i.e. discount with at least one range with quantity threshold greater than one).
end_date	xsd:datetime	false	true	Discount end date
origin_price	xsd:decimal	false	true	Origin price
ranges	range [zero or N]	false	false	Discount ranges
price	xsd:decimal	true	false	Price of the offer if the quantity threshold is reached
quantity_threshold	xsd:int	true	false	Minimum quantity to order for the given price to be applied
start_date	xsd:datetime	false	true	Discount start date
favorite_rank	xsd:int	false	false	Favorite rank if the offer is favorite and if the favorite offer feature is enabled
logistic_class	logistic_class	true	true	The logistic class associated to the offer
code	xsd:string	true	false	The code of the logistic class
label	xsd:string	true	false	The label of the logistic class
min_quantity_alert	xsd:int	true	false	
min_shipping_price	xsd:decimal	true	false	Unit price of the min shipping rate
min_shipping_price_additional	xsd:decimal	true	false	Additional unit price associated to the min shipping rate
min_shipping_type	xsd:string	true	false	Code of the shipping type associated to the min shipping rate
min_shipping_zone	xsd:string	true	false	Code of the shipping zone associated to the min shipping rate
offer_additional_fields	offer_additional_field[zero or N]	false	false	List of additional fields
code	xsd:string	true	false	Code of the additional field
value	xsd:string	true	false	Value of the additional field
offer_id	xsd:long	true	false	Identifier of the offer

price	xsd:decimal	true	false	Price of the offer
price_additional_info	xsd:string	false	true	Price additional information of the offer
product_references	product_reference[zero or N]	true	false	List of product's references
reference	xsd:string	true	false	Value of the reference
reference_type	xsd:string	true	false	Type of the reference
product_sku	xsd:string	true	false	Sku of the product
product_title	xsd:string	true	false	Title of the product
quantity	xsd:int	true	false	Available quantity of the offer
shop_sku	xsd:string	true	false	Shop's sku of the offer
state_code	xsd:string	true	false	State of the offer
total_price	xsd:decimal	true	false	Total price of the offer (price + min shipping rate)

PRODUCES:

application/json
application/xml

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/offers/{offer}

OF24 - Create, update or delete offers**METHOD:** POST**URL STRUCTURE:** /api/offers

Returns the import identifier to track the status of the update.

Recommended: 12/hour**Maximum:** 60/hour**QUERY PARAMETERS:** N/A**INPUT:**

OffersBean - [required] List of representation of offers

JSON:

```
{  
    "offers": [  
        {  
            "available_ended": Date,  
            "available_started": Date,  
            "description": String,  
            "discount": {  
                "end_date": Date,  
                "price": Number,  
                "ranges": [  
                    {  
                        "price": Number,  
                        "quantity_threshold": Number  
                    }  
                ],  
                "start_date": Date  
            },  
            "internal_description": String,  
            "logistic_class": String,  
            "min_quantity_alert": Number,  
            "offer_additional_fields": [  
                {  
                    "code": String,  
                    "value": String  
                }  
            ],  
            "price": Number,  
            "price_additional_info": String,  
            "product_id": String,  
            "product_id_type": String,  
            "quantity": Number,  
            "shop_sku": String,  
            "state_code": String,  
            "update_delete": String  
        }  
    ]  
}
```

XML:

```

<body>
  <offers>
    <offer> [zero or N]
      <available_ended>xsd:datetime</available_ended>
      <available_started>xsd:datetime</available_started>
      <description>xsd:string</description>
      <discount>
        <end_date>xsd:datetime</end_date>
        <price>xsd:decimal</price>
        <ranges>
          <range> [zero or N]
            <price>xsd:decimal</price>
            <quantity_threshold>xsd:int</quantity_threshold>
          </range>
        </ranges>
        <start_date>xsd:datetime</start_date>
      </discount>
      <internal_description>xsd:string</internal_description>
      <logistic_class>xsd:string</logistic_class>
      <min_quantity_alert>xsd:int</min_quantity_alert>
      <offer_additional_fields>
        <offer_additional_field> [zero or N]
          <code>xsd:string</code>
          <value>xsd:string</value>
        </offer_additional_field>
      </offer_additional_fields>
      <price>xsd:decimal</price>
      <price_additional_info>xsd:string</price_additional_info>
      <product_id>xsd:string</product_id>
      <product_id_type>xsd:string</product_id_type>
      <quantity>xsd:int</quantity>
      <shop_sku>xsd:string</shop_sku>
      <state_code>xsd:string</state_code>
      <update_delete>xsd:string</update_delete>
    </offer>
  </offers>
</body>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
offers	offer [zero or N]	true	false	
available_ended	xsd:datetime	true	true	End date of the period of availability

available_started	xsd:datetime	true	true	Started date of the period of availability
description	xsd:string	true	true	Description of the offer
discount	discount	false	true	Discount of an offer
end_date	xsd:datetime	false	true	Discount end date
price	xsd:decimal	false	true	Discount price. Not to be used when ranges are specified.
ranges	range [zero or N]	false	false	Discount ranges
price	xsd:decimal	true	false	Price of the offer if the quantity threshold is reached
quantity_threshold	xsd:int	true	false	Minimum quantity to order for the given price to be applied
start_date	xsd:datetime	false	true	Discount start date
internal_description	xsd:string	true	true	Internal description of the offer
logistic_class	xsd:string	false	false	
min_quantity_alert	xsd:int	true	true	Minimum quantity of the offer
offer_additional_fields	offer_additional_field[zero or N]	false	false	Additional fields of the offer
code	xsd:string	true	false	Code of the additional field
value	xsd:string	true	false	Value of the additional field
price	xsd:decimal	true	false	Price of the offer
price_additional_info	xsd:string	false	true	Price additional information of the offer
product_id	xsd:string	true	true	Reference of the product associated with the offer
product_id_type	xsd:string	true	true	Reference type of the product associated with the offer
quantity	xsd:int	true	false	Quantity available of the offer
shop_sku	xsd:string	true	false	Sku of the offer
state_code	xsd:string	true	false	State of the offer

update_delete	xsd:string	true	true	Update delete flag. Could be empty (means "update"), "update" or "delete".
---------------	------------	------	------	--

OUTPUT:

OfferImportTrackingBean - Import identifier

Offer import tracking information

JSON:

```
{
  "import_id": Number,
  "product_import_id": Number
}
```

XML:

```
<offer_import_tracking>
  <import_id>xsd:long</import_id>
  <product_import_id>xsd:long</product_import_id>
</offer_import_tracking>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
import_id	xsd:long	true	false	Identifier of the offer import
product_import_id	xsd:long	false	false	Identifier of the product import

PRODUCES:

application/json

application/xml

CONSUMES:

application/json

application/xml

HTTP RETURN CODES: 201 – Success**HTTP RESPONSE HEADERS:****Location** - URI of the new import.**HTTP Example:** POST /api/offers

OF61 - Get the list of offer states**METHOD:** GET**URL STRUCTURE:** /api/offers/states**Recommended:** 1/day**Maximum:** 1/day**QUERY PARAMETERS:** N/A**SORT:****sortIndex** - (default) Sort by index set in the back-office**OUTPUT:**

OfferStatesBean - Representation of an offer

List of offer states representations without pagination

JSON:

```
{  
    "offer_states": [ {  
        "code": String,  
        "label": String  
    } ]  
}
```

XML:

```
<body>  
    <offer_states>  
        <offer_state> [zero or N]  
            <code>xsd:string</code>  
            <label>xsd:string</label>  
        </offer_state>  
    </offer_states>  
</body>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
offer_states	offer_state [zero or N]	true	false	List of offer states representations
code	xsd:string	true	false	Code of the offer state

label	xsd:string	true	false	Label of the offer state
-------	------------	------	-------	--------------------------

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/offers/states

OR11 - List orders**METHOD:** GET**URL STRUCTURE:** /api/orders**Recommended:**

- Asynchronous: 12/hour
- Synchronous: On each order page view

Maximum:

- Asynchronous: 60/hour
- Synchronous: On each order page view.

Note: this resource supports pagination (see documentation)

QUERY PARAMETERS:

order_ids - [optional] A list of identifiers of an order.

order_state_codes - [optional] List of order state code.

channel_codes - [optional] A list of channel code.

start_date - [optional] creation date for filtering. Format : "yyyy-MM-dd'T'HH:mm:ss"

end_date - [optional] creation date for filtering. Format : "yyyy-MM-dd'T'HH:mm:ss"

start_update_date - [optional] last update date for filtering. Format : "yyyy-MM-dd'T'HH:mm:ss"

end_update_date - [optional] last update date for filtering. Format : "yyyy-MM-dd'T'HH:mm:ss"

paginate - [optional] Control the pagination usage. Default : true

SORT:

dateCreated - (default) Sort by creation date, order identifier, shop name and then by index of the order line

OUTPUT:

ShopOrdersDto - List of orders

List of orders with pagination

JSON:

```
{  
  "orders": [  
    {  
      "acceptance_decision_date": Date,  
      "channel": {  
        "code": String,  
        "label": String  
      },  
      "commercial_id": String,  
      "customer": {  
        "id": String,  
        "name": String  
      },  
      "line_items": [  
        {  
          "id": String,  
          "item": {  
            "id": String,  
            "name": String  
          },  
          "quantity": Number  
        }  
      ],  
      "order": {  
        "id": String,  
        "status": String  
      },  
      "shop": {  
        "id": String,  
        "name": String  
      }  
    }  
  ]  
}
```

```
"created_date": Date,  
"currency_iso_code": "CAD",  
"customer": {  
    "billing_address": {  
        "city": String,  
        "civility": String,  
        "company": String,  
        "country": String,  
        "country_iso_code": "CAN",  
        "firstname": String,  
        "lastname": String,  
        "phone": String,  
        "phone_secondary": String,  
        "state": String,  
        "street_1": String,  
        "street_2": String,  
        "zip_code": String  
    },  
    "civility": String,  
    "customer_id": String,  
    "firstname": String,  
    "lastname": String,  
    "locale": String,  
    "shipping_address": {  
        "additional_info": String,  
        "city": String,  
        "civility": String,  
        "company": String,  
        "country": String,  
        "country_iso_code": "CAN",  
        "firstname": String,  
        "lastname": String,  
        "phone": String,  
        "phone_secondary": String,  
        "state": String,  
        "street_1": String,  
        "street_2": String,  
        "zip_code": String  
    }  
},  
"has_customer_message": Boolean,  
"has_incident": Boolean,  
"has_invoice": Boolean,  
"last_updated_date": Date,  
"order_id": String,  
"order_lines": [  
    "can_refund": Boolean,  
    "category_code": String,  
    "category_label": String,
```

```
"commission_fee": Number,
"commission_taxes": [
    "amount": Number,
    "code": String,
    "rate": Number
],
"created_date": Date,
"debited_date": Date,
"description": String,
"last_updated_date": Date,
"offer_id": Number,
"offer_sku": String,
"offer_state_code": String,
"order_line_additional_fields": [
    {
        "code": String,
        "value": String
    }
],
"order_line_id": String,
"order_line_index": Number,
"order_line_state": "STAGING" | "WAITING_ACCEPTANCE" | "WAITING_DEBIT"
| "WAITING_DEBIT_PAYMENT" | "SHIPPING" | "SHIPPED" | "TO_COLLECT" |
"RECEIVED" | "CLOSED" | "REFUSED" | "CANCELED" | "INCIDENT_OPEN" |
"REFUNDED",
"order_line_state_reason_code": String,
"order_line_state_reason_label": String,
"price": Number,
"price_additional_info": String,
"price_unit": Number,
"product_sku": String,
"product_title": String,
"quantity": Number,
"received_date": Date,
"refunds": [
    {
        "amount": Number,
        "commission_amount": Number,
        "commission_taxes": [
            {
                "amount": Number,
                "code": String
            }
        ],
        "commission_total_amount": Number,
        "created_date": Date,
        "id": String,
        "quantity": Number,
        "reason_code": String,
        "shipping_amount": Number,
        "shipping_taxes": [
            {
                "amount": Number,
                "code": String
            }
        ],
        "total_amount": Number
    }
]
}
```

```

    "state": "WAITING_REFUND" | "WAITING_REFUND_PAYMENT" | "REFUNDED",
    "taxes": [
        {
            "amount": Number,
            "code": String
        }
    ],
    "transaction_date": Date,
    "transaction_number": String
],
    "shipped_date": Date,
    "shipping_price": Number,
    "shipping_taxes": [
        {
            "amount": Number,
            "code": String
        }
    ],
    "taxes": [
        {
            "amount": Number,
            "code": String
        }
    ],
    "total_commission": Number,
    "total_price": Number
},
    "order_state": "STAGING" | "WAITING_ACCEPTANCE" | "WAITING_DEBIT" |
"WAITING_DEBIT_PAYMENT" | "SHIPPING" | "SHIPPED" | "TO_COLLECT" | "RECEIVED"
| "CLOSED" | "REFUSED" | "CANCELED",
    "order_state_reason_code": String,
    "order_state_reason_label": String,
    "payment_type": String,
    "price": Number,
    "shipping_company": String,
    "shipping_price": Number,
    "shipping_tracking": String,
    "shipping_tracking_url": String,
    "shipping_type_code": String,
    "shipping_type_label": String,
    "shipping_zone_code": String,
    "shipping_zone_label": String,
    "total_commission": Number,
    "total_price": Number
},
    "total_count": Number
}

```

XML:

```

<body>
    <orders>
        <order> [zero or N]
            <acceptance_decision_date>xsd:datetime</acceptance_decision_date>
            <channel>
                <code>xsd:string</code>

```

```
<label>xsd:string</label>
</channel>
<commercial_id>xsd:string</commercial_id>
<created_date>xsd:datetime</created_date>
<currency_iso_code>"CAD"</currency_iso_code>
<customer>
  <billing_address>
    <city>xsd:string</city>
    <civility>xsd:string</civility>
    <company>xsd:string</company>
    <country>xsd:string</country>
    <country_iso_code>"CAN"</country_iso_code>
    <firstname>xsd:string</firstname>
    <lastname>xsd:string</lastname>
    <phone>xsd:string</phone>
    <phone_secondary>xsd:string</phone_secondary>
    <state>xsd:string</state>
    <street_1>xsd:string</street_1>
    <street_2>xsd:string</street_2>
    <zip_code>xsd:string</zip_code>
  </billing_address>
  <civility>xsd:string</civility>
  <customer_id>xsd:string</customer_id>
  <firstname>xsd:string</firstname>
  <lastname>xsd:string</lastname>
  <locale>xsd:string</locale>
  <shipping_address>
    <additional_info>xsd:string</additional_info>
    <city>xsd:string</city>
    <civility>xsd:string</civility>
    <company>xsd:string</company>
    <country>xsd:string</country>
    <country_iso_code>"CAN"</country_iso_code>
    <firstname>xsd:string</firstname>
    <lastname>xsd:string</lastname>
    <phone>xsd:string</phone>
    <phone_secondary>xsd:string</phone_secondary>
    <state>xsd:string</state>
    <street_1>xsd:string</street_1>
    <street_2>xsd:string</street_2>
    <zip_code>xsd:string</zip_code>
  </shipping_address>
</customer>
<has_customer_message>xsd:boolean</has_customer_message>
<has_incident>xsd:boolean</has_incident>
<has_invoice>xsd:boolean</has_invoice>
<last_updated_date>xsd:datetime</last_updated_date>
<order_id>xsd:string</order_id>
<order_lines>
```

```
<order_line> [zero or N]
  <can_refund>xsd:boolean</can_refund>
  <category_code>xsd:string</category_code>
  <category_label>xsd:string</category_label>
  <commission_fee>xsd:decimal</commission_fee>
  <commission_taxes>
    <commission_tax> [zero or N]
      <amount>xsd:decimal</amount>
      <code>xsd:string</code>
      <rate>xsd:decimal</rate>
    </commission_tax>
  </commission_taxes>
  <created_date>xsd:datetime</created_date>
  <debited_date>xsd:datetime</debited_date>
  <description>xsd:string</description>
  <last_updated_date>xsd:datetime</last_updated_date>
  <offer_id>xsd:long</offer_id>
  <offer_sku>xsd:string</offer_sku>
  <offer_state_code>xsd:string</offer_state_code>
  <order_line_additional_fields>
    <order_line_additional_field> [zero or N]
      <code>xsd:string</code>
      <value>xsd:string</value>
    </order_line_additional_field>
  </order_line_additional_fields>
  <order_line_id>xsd:string</order_line_id>
  <order_line_index>xsd:int</order_line_index>
  <order_line_state>"STAGING" | "WAITING_ACCEPTANCE" |
"WAITING_DEBIT" | "WAITING_DEBIT_PAYMENT" | "SHIPPING" | "SHIPPED" |
"TO_COLLECT" | "RECEIVED" | "CLOSED" | "REFUSED" | "CANCELED" |
"INCIDENT_OPEN" | "REFUNDED"</order_line_state>
    <order_line_state_reason_code>xsd:string</order_line_state_reason_code>
    <order_line_state_reason_label>xsd:string</order_line_state_reason_label>
    <price>xsd:decimal</price>
    <price_additional_info>xsd:string</price_additional_info>
    <price_unit>xsd:decimal</price_unit>
    <product_sku>xsd:string</product_sku>
    <product_title>xsd:string</product_title>
    <quantity>xsd:int</quantity>
    <received_date>xsd:datetime</received_date>
    <refunds>
      <refund> [zero or N]
        <amount>xsd:decimal</amount>
        <commission_amount>xsd:decimal</commission_amount>
        <commission_taxes>
          <commission_tax> [zero or N]
            <amount>xsd:decimal</amount>
```

```
<code>xsd:string</code>
</commission_tax>
</commission_taxes>
<commission_total_amount>xsd:decimal</commission_total_amount>
<created_date>xsd:datetime</created_date>
<id>xsd:string</id>
<quantity>xsd:int</quantity>
<reason_code>xsd:string</reason_code>
<shipping_amount>xsd:decimal</shipping_amount>
<shipping_taxes>
    <shipping_tax> [zero or N]
        <amount>xsd:decimal</amount>
        <code>xsd:string</code>
    </shipping_tax>
</shipping_taxes>
<state>"WAITING_REFUND" | "WAITING_REFUND_PAYMENT" |
"REFUNDED"</state>
<taxes>
    <tax> [zero or N]
        <amount>xsd:decimal</amount>
        <code>xsd:string</code>
    </tax>
</taxes>
<transaction_date>xsd:datetime</transaction_date>
<transaction_number>xsd:string</transaction_number>
</refund>
</refunds>
<shipped_date>xsd:datetime</shipped_date>
<shipping_price>xsd:decimal</shipping_price>
<shipping_taxes>
    <shipping_tax> [zero or N]
        <amount>xsd:decimal</amount>
        <code>xsd:string</code>
    </shipping_tax>
</shipping_taxes>
<taxes>
    <tax> [zero or N]
        <amount>xsd:decimal</amount>
        <code>xsd:string</code>
    </tax>
</taxes>
<total_commission>xsd:decimal</total_commission>
<total_price>xsd:decimal</total_price>
</order_line>
</order_lines>
<order_state>"STAGING" | "WAITING_ACCEPTANCE" | "WAITING_DEBIT" |
"WAITING_DEBIT_PAYMENT" | "SHIPPING" | "SHIPPED" | "TO_COLLECT" | "RECEIVED" |
"CLOSED" | "REFUSED" | "CANCELED"</order_state>
<order_state_reason_code>xsd:string</order_state_reason_code>
```

```

<order_state_reason_label>xsd:string</order_state_reason_label>
<payment_type>xsd:string</payment_type>
<price>xsd:decimal</price>
<shipping_company>xsd:string</shipping_company>
<shipping_price>xsd:decimal</shipping_price>
<shipping_tracking>xsd:string</shipping_tracking>
<shipping_tracking_url>xsd:string</shipping_tracking_url>
<shipping_type_code>xsd:string</shipping_type_code>
<shipping_type_label>xsd:string</shipping_type_label>
<shipping_zone_code>xsd:string</shipping_zone_code>
<shipping_zone_label>xsd:string</shipping_zone_label>
<total_commission>xsd:decimal</total_commission>
<total_price>xsd:decimal</total_price>
</order>
</orders>
<total_count>xsd:int</total_count>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
orders	order [zero or N]	true	false	List of orders
acceptance_decision_date	xsd:datetime	true	true	The date where the shop decided to accept or refuse the order
channel	channel	false	true	The channel of the commercial order
code	xsd:string	true	false	The code of the channel
label	xsd:string	true	false	The label of the channel
commercial_id	xsd:string	true	false	Identifier of the order
created_date	xsd:datetime	true	false	Creation date of the order
currency_iso_code	"CAD"	true	true	The currency of the shop (iso format)
customer	customer	true	false	Information of the customer who ordered
billing_address	billing_address	true	true	Billing address of the customer
city	xsd:string	true	false	City of the address
civility	xsd:string	true	false	Civility of the person associated with the address

company	xsd:string	false	true	Company of the person associated with the address
country	xsd:string	true	false	Country of the address
country_iso_code	"CAN"	false	true	Country ISO 3166-1 code of the address
firstname	xsd:string	true	false	Firstname of the person associated with the address
lastname	xsd:string	true	false	Lastname of the person associated with the address
phone	xsd:string	true	false	Phone
phone_secondary	xsd:string	true	false	Phone secondary
state	xsd:string	false	true	State of the address
street_1	xsd:string	true	false	First information line of the street of the address
street_2	xsd:string	false	true	Second information line of the street of the address
zip_code	xsd:string	true	false	ZipCode of the address
civility	xsd:string	false	false	
customer_id	xsd:string	true	false	Identifier of the customer
firstname	xsd:string	true	false	Firstname of the customer
lastname	xsd:string	true	false	Lastname of the customer
locale	xsd:string	false	true	RFC 3066 code of the customer's locale
shipping_address	shipping_address	true	true	Shipping address of the customer
additional_info	xsd:string	false	true	Additional information for the shipping address
city	xsd:string	true	false	City of the address
civility	xsd:string	true	false	Civility of the person associated with the address

company	xsd:string	false	true	Company of the person associated with the address
country	xsd:string	true	false	Country of the address
country_iso_code	"CAN"	false	true	Country ISO 3166-1 code of the address
firstname	xsd:string	true	false	Firstname of the person associated with the address
lastname	xsd:string	true	false	Lastname of the person associated with the address
phone	xsd:string	true	false	Phone
phone_secondary	xsd:string	true	false	Phone secondary
state	xsd:string	false	true	State of the address
street_1	xsd:string	true	false	First information line of the street of the address
street_2	xsd:string	false	true	Second information line of the street of the address
zip_code	xsd:string	true	false	ZipCode of the address
has_customer_message	xsd:boolean	true	false	Indicate if the customer has send a message related to this order
has_incident	xsd:boolean	true	false	Indicate if order has incident on at least one order line
has_invoice	xsd:boolean	true	false	Is an invoice available for this order.
last_updated_date	xsd:datetime	true	true	Last update date of the order
order_id	xsd:string	true	false	Identifier of the order
order_lines	order_line [zero or N]	true	false	Lines of the order
can_refund	xsd:boolean	true	true	Indicate if the order line can be refunded by the operator
category_code	xsd:string	true	false	Category code of the product associated with the line
category_label	xsd:string	true	false	Category label of the product associated with the line

commission_fee	xsd:decimal	true	false	Commission fee of the order line.
commission_taxes	commission_tax [zero or N]	false	false	Commission taxes
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
rate	xsd:decimal	true	false	The rate of the tax, as a percentage (between 0 and 100).
created_date	xsd:datetime	true	false	Creation date of the line
debited_date	xsd:datetime	true	true	Debited date of the order line
description	xsd:string	true	true	Description of the line
last_updated_date	xsd:datetime	true	true	Date of the last update of the line
offer_id	xsd:long	true	false	Id of the associated offer
offer_sku	xsd:string	true	false	Sku of the associated offer (this is the Shop SKU)
offer_state_code	xsd:string	true	false	State of the offer associated with the line
order_line_additional_fields	order_line_additional_field [zero or N]	true	false	List of additional fields
code	xsd:string	true	false	Code of the additional field
value	xsd:string	true	false	Value of the additional field
order_line_id	xsd:string	true	false	Identifier of the line
order_line_index	xsd:int	true	false	Index of the line in the order
order_line_state	"STAGING" "WAITING_ACCEPTANCE" "WAITING_DEBIT" "WAITING_DEBIT_PAYMENT" "SHIPPING" "SHIPPED" "TO_COLLECT" "RECEIVED" "CLOSED" "REFUSED" "CANCELED" "INCIDENT_OPEN" "REFUNDED"	true	false	State of the line
order_line_state_reason_code	xsd:string	true	true	Code of the reason of the line state

order_line_state_reason_label	xsd:string	true	true	Label of the reason of the line state
price	xsd:decimal	true	false	Price of the line without shipping price
price_additional_info	xsd:string	false	true	Price additional information of the offer
price_unit	xsd:decimal	true	false	Unit price for the offer associated with the line
product_sku	xsd:string	true	false	Sku of the product associated with the line
product_title	xsd:string	true	false	Title of the product associated with the line
quantity	xsd:int	true	false	Quantity of product for the line
received_date	xsd:datetime	true	true	Date of receipt of the product
refunds	refund [zero or N]	true	false	List of the refunds
amount	xsd:decimal	false	true	The amount of the refund
commission_amount	xsd:decimal	true	false	The commission amount of the refund
commission_taxes	commission_tax [zero or N]	true	false	The commission taxes of the refund
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
commission_total_amount	xsd:decimal	true	false	The total commission amount of the refund (commission amount + commission taxes)
created_date	xsd:datetime	true	false	The creation date of the refund
id	xsd:string	true	false	The id of the refund
quantity	xsd:int	false	true	The quantity of products refunded (This quantity is informative only)
reason_code	xsd:string	true	false	The code of the refund reason
shipping_amount	xsd:decimal	false	true	The shipping amount of the refund
shipping_taxes	shipping_tax [zero or N]	false	true	The taxes on the shipping price

amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
state	"WAITING_REFUND" "WAITING_REFUND_PAYMENT" "REFUNDED"	true	false	The state of the refund
taxes	tax [zero or N]	false	true	The taxes on the price
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
transaction_date	xsd:datetime	true	false	The transaction date of the refund payment
transaction_number	xsd:string	true	false	The transaction number of the refund payment
shipped_date	xsd:datetime	true	true	Shipped date of the order line
shipping_price	xsd:decimal	true	false	Total price of the shipping price of the line
shipping_taxes	shipping_tax [zero or N]	false	false	List of taxes applied on shipping charges
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
taxes	tax [zero or N]	false	false	List of taxes applied on product price
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
total_commission	xsd:decimal	true	false	Total commission of the order line (sum of the commission fee and the commission vat).
total_price	xsd:decimal	true	false	Price of the line with shipping price
order_state	"STAGING" "WAITING_ACCEPTANCE" "WAITING_DEBIT" "WAITING_DEBIT_PAYMENT" "SHIPPING" "SHIPPED" "TO_COLLECT" "RECEIVED"	true	false	State of the order. States REFUSED, INCIDENT_OPEN, INCIDENT_CLOSED, WAITING_REFUND, WAITING_REFUND_PAYMENT are

	"CLOSED" "REFUSED" "CANCELED"			exclusively for order lines (never on orders)
order_state_reason_code	xsd:string	true	true	Code of the reason of the order state
order_state_reason_label	xsd:string	true	true	Label of the reason of the order state
payment_type	xsd:string	true	true	The type of payment used by the customer to pay this order
price	xsd:decimal	true	false	Price of the order (sum of order line price)
shipping_company	xsd:string	false	true	Shipping company
shipping_price	xsd:decimal	true	false	Shipping price of the order (sum of order line shipping price)
shipping_tracking	xsd:string	false	true	Shipping tracking
shipping_tracking_url	xsd:string	false	true	Shipping tracking url
shipping_type_code	xsd:string	true	false	Code of shipping's type
shipping_type_label	xsd:string	true	false	Label of shipping's type
shipping_zone_code	xsd:string	true	false	Code of shipping's zone
shipping_zone_label	xsd:string	true	false	Label of shipping's zone
total_commission	xsd:decimal	true	false	Total commission of the order (sum of the order line total commission)
total_price	xsd:decimal	true	false	Total price of the order (sum of the order price and the order shipping price)
total_count	xsd:int	true	false	Total count of orders

PRODUCES:

application/json
application/xml

HTTP RETURN CODES: 200 – Success

HTTP Example: GET

```
/api/orders?order_ids=...&order_state_codes=...&channel_codes=...&start_date=...&end_<br/>date=...&start_update_date=...&end_update_date=...&paginate=...
```

OR21 - Accept or refuse order lines of an order which are in "WAITING_ACCEPTANCE" state

METHOD: PUT

URL STRUCTURE: /api/orders/{order_id}/accept

order_id - [required] The identifier of the order

Recommended: On each new order

Maximum: On each new order

QUERY PARAMETERS: N/A

INPUT:

OrderAcceptBean - [required] The representation of the order line states
Representation of an order for acceptance

JSON:

```
{
  "order_lines": [
    {
      "accepted": Boolean,
      "id": String
    }
  ]
}
```

XML:

```
<order>
  <order_lines>
    <order_line> [zero or N]
      <accepted>xsd:boolean</accepted>
      <id>xsd:string</id>
    </order_line>
  </order_lines>
</order>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
order_lines	order_line [zero or N]	true	false	List of representations of order line for acceptance
accepted	xsd:boolean	true	false	Boolean indicates if the order line is accepted or refused
id	xsd:string	true	false	Identifier of the order line

OUTPUT:

Nothing

PRODUCES:

application/json

application/xml

CONSUMES:

application/json

application/xml

HTTP RETURN CODES: 204 – Success

HTTP Example: `PUT /api/orders/{order_id}/accept`

OR23 - Update carrier tracking information of a given order

If the carrier is not registered, the complete tracking url can be provided.

METHOD: PUT

URL STRUCTURE: /api/orders/{order_id}/tracking

order_id - [required] the identifier of the order.

Recommended : On each order tracking info update

Maximum : On each order tracking info update

QUERY PARAMETERS: N/A

INPUT:

TrackingInfoBean - [required] the bean containing new tracking informations.

This bean contains information required to track shipment of an order.

JSON:

```
{
  "carrier_code": String,
  "carrier_name": String,
  "carrier_url": String,
  "tracking_number": String
}
```

XML:

```
<tracking>
  <carrier_code>xsd:string</carrier_code>
  <carrier_name>xsd:string</carrier_name>
  <carrier_url>xsd:string</carrier_url>
  <tracking_number>xsd:string</tracking_number>
</tracking>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
carrier_code	xsd:string	false	false	The code of a carrier. This information is mandatory for a registered carrier.
carrier_name	xsd:string	false	false	The name of a carrier. This information is mandatory for an unregistered carrier.

carrier_url	xsd:string	false	false	The tracking url of a carrier. This information is unused for registered carriers (because computed automatically). This information is optional for unregistered carriers.
tracking_number	xsd:string	false	false	The carrier tracking number. This information is mandatory for a registered carrier with a URL requiring a tracking number.

OUTPUT:

Nothing

PRODUCES:

application/json
application/xml

CONSUMES:

application/json
application/xml

HTTP RETURN CODES: 204 – No Content

HTTP Example: PUT /api/orders/{order_id}/tracking

OR24 - Update the shipment of the order which is in "SHIPPING" state to "SHIPPED" state

METHOD: PUT

URL STRUCTURE: /api/orders/{order_id}/ship

order_id - [required] The identifier of the order

Recommended: On each shop order shipment

Maximum: On each shop order shipment

QUERY PARAMETERS: N/A

OUTPUT:

Nothing

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 204 – Success

HTTP Example: PUT /api/orders/{order_id}/ship

OR28 - Demand refunds on order lines

METHOD: PUT**URL STRUCTURE:** /api/orders/refund**Recommended:** On each new refund demand**Maximum:** On each new refund demand**QUERY PARAMETERS:** N/A**INPUT:**

RefundsDemandBean - [required] The list refunds to be created
List of representations refunds to be created

JSON:

```
{  
  "refunds": [  
    {  
      "amount": Number,  
      "currency_iso_code": "CAD",  
      "order_line_id": String,  
      "quantity": Number,  
      "reason_code": String,  
      "shipping_amount": Number,  
      "shipping_taxes": [  
        {  
          "amount": Number,  
          "code": String  
        },  
        {  
          "amount": Number,  
          "code": String  
        }  
      ]  
    }  
  ]  
}
```

XML:

```
<body>  
<refunds>  
  <refund> [zero or N]  
    <amount>xsd:decimal</amount>  
    <currency_iso_code>"CAD"</currency_iso_code>  
    <order_line_id>xsd:string</order_line_id>  
    <quantity>xsd:int</quantity>  
    <reason_code>xsd:string</reason_code>  
    <shipping_amount>xsd:decimal</shipping_amount>  
    <shipping_taxes>  
      <shipping_tax> [zero or N]
```

```

<amount>xsd:decimal</amount>
<code>xsd:string</code>
</shipping_tax>
</shipping_taxes>
<taxes>
  <tax> [zero or N]
    <amount>xsd:decimal</amount>
    <code>xsd:string</code>
  </tax>
</taxes>
</refund>
</refunds>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
refunds	refund [zero or N]	true	false	List of representations of refunds to be created
amount	xsd:decimal	true	false	The offer part amount to be refunded.
currency_iso_code	"CAD"	false	false	The currency used by the shop (iso format) The field is required if the shop is not using the default Operator platform currency
order_line_id	xsd:string	true	false	The identifier of the order line that must be refunded
quantity	xsd:int	false	true	The quantity of products to refund. This quantity is informative only.
reason_code	xsd:string	true	false	The reason code of the refund
shipping_amount	xsd:decimal	true	false	The shipping charges part amount to be refunded.
shipping_taxes	shipping_tax [zero or N]	false	true	The taxes to be refunded on the shipping price This field is required if the order has shipping taxes
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
taxes	tax [zero or N]	false	true	The taxes to be refunded on the product price This field is required if the order has taxes
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax

OUTPUT:

CreatedRefundsBean - Representation of created refunds
List of representations of created refunds

JSON:

```
{  
    "refunds": [  
        {  
            "amount": Number,  
            "currency_iso_code": "CAD",  
            "order_line_id": String,  
            "quantity": Number,  
            "reason_code": String,  
            "refund_id": String,  
            "shipping_amount": Number,  
            "shipping_taxes": [  
                {  
                    "amount": Number,  
                    "code": String  
                }],  
            "taxes": [  
                {  
                    "amount": Number,  
                    "code": String  
                }]  
        }]  
    }  
}
```

XML:

```
<body>  
    <refunds>  
        <refund> [zero or N]  
            <amount>xsd:decimal</amount>  
            <currency_iso_code>"CAD"</currency_iso_code>  
            <order_line_id>xsd:string</order_line_id>  
            <quantity>xsd:int</quantity>  
            <reason_code>xsd:string</reason_code>  
            <refund_id>xsd:string</refund_id>  
            <shipping_amount>xsd:decimal</shipping_amount>  
            <shipping_taxes>  
                <shipping_tax> [zero or N]  
                    <amount>xsd:decimal</amount>  
                    <code>xsd:string</code>  
                </shipping_tax>  
            </shipping_taxes>  
            <taxes>  
                <tax> [zero or N]  
                    <amount>xsd:decimal</amount>  
                    <code>xsd:string</code>  
                </tax>  
            </taxes>  
        </refund>  
    </refunds>  
</body>
```

```

</taxes>
</refund>
</refunds>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
refunds	refund [zero or N]	true	false	List of representations of created refunds
amount	xsd:decimal	true	false	The offer part amount of the refund.
currency_iso_code	"CAD"	false	false	The currency used by the shop (iso format) The field is required if the shop is not using the default Operator platform currency
order_line_id	xsd:string	true	false	The identifier of the order line on which the refund has been created.
quantity	xsd:int	false	true	The quantity of goods to refund
reason_code	xsd:string	true	false	The reason code of the refund.
refund_id	xsd:string	true	false	The identifier of the created refund.
shipping_amount	xsd:decimal	true	false	The shipping charges part amount of the refund.
shipping_taxes	shipping_tax [zero or N]	false	false	The amounts of shipping taxes to refund
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax
taxes	tax [zero or N]	false	true	The amounts of price taxes to refund
amount	xsd:decimal	true	false	The amount of the tax
code	xsd:string	true	false	The code of the tax

PRODUCES:

application/json
application/xml

CONSUMES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success

HTTP Example: PUT /api/orders/refund

OR72 - Lists order's documents

METHOD: GET**URL STRUCTURE:** /api/orders/documents

Returns a list of all the documents available on the order.

Recommended: On each documents list page view**Maximum:** On each documents list page view**QUERY PARAMETERS:****order_ids** - [required] the orders' identifiers, using comma as separator**OUTPUT:**

OrderDocumentsBean - List of documents

JSON:

```
{  
  "order_documents": [  
    {"date_uploaded": Date,  
     "file_name": String,  
     "file_size": Number,  
     "id": Number,  
     "order_id": String,  
     "type": String  
   ]  
}
```

XML:

```
<body>  
  <order_documents>  
    <order_document> [zero or N]  
      <date_uploaded>xsd:datetime</date_uploaded>  
      <file_name>xsd:string</file_name>  
      <file_size>xsd:long</file_size>  
      <id>xsd:long</id>  
      <order_id>xsd:string</order_id>  
      <type>xsd:string</type>  
    </order_document>  
  </order_documents>  
</body>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
order_documents	order_document[zero or N]	true	false	The document details
date_uploaded	xsd:datetime	false	false	The upload date of the document
file_name	xsd:string	true	false	The document file name
file_size	xsd:long	false	false	
id	xsd:long	true	false	The identifier of the document
order_id	xsd:string	true	false	The id of the order owning the document
type	xsd:string	true	false	The Shop Document Type code of the document

PRODUCES:

application/json

application/xml

HTTP Example: GET /api/orders/documents?order_ids=...

OR73 - Download one or multiple documents associated to one or multiple orders

METHOD: GET

URL STRUCTURE: /api/orders/documents/download

Recommended: On each documents download

Maximum: On each documents download.

QUERY PARAMETERS:

order_ids - A list of identifiers of the orders (separated by a comma)

document_ids - A list of document identifiers (separated by a comma)

document_codes - [optional] A list of document type codes (separated by a comma)

- If a list of document identifiers is specified only these documents are downloaded.
 - If more than one document id is specified, the documents will be wrapped in a ZIP archive
 - If only one document id is specified the document will not be zipped
 - If a list of order identifiers is specified, all documents from those orders are downloaded.
- Use a list of order document type codes to retrieve specific types from those orders. In this case, the output of the API will always be a ZIP archive even if there is only one document to retrieve.

When documents are retrieved, they're wrapped into a ZIP archive except when only one document id is specified. The tree structure of this archive is as follows:

```
documents-timestamp.zip
|
|__ order_id_x/
|   |__ foo.txt
|   |__ bar.txt
|   |__ baz.pdf
|
|__ order_id_y/
|   |__ image.png
|   |__ image(1).png
```

OUTPUT:

Binary

PRODUCES:

application/json

application/xml

HTTP RETURN CODES:

200 – Success

404 – if at least one document id, one order id or one document type code is invalid

HTTP Example: GET

/api/orders/documents/download?order_ids=...&document_ids=...&document_codes=...

OR74 - Upload documents to associate to an order**METHOD:** POST**URL STRUCTURE:** /api/orders/{order_id}/documents**order_id** - [required] The identifier of the order

Documents filenames must be distinct. Only documents of the following types are supported:
csv, doc, xls, ppt, pdf, odt, ods, odp, txt, rtf, png, jpg, gif.

An order can have a maximum of 50 documents.

Recommended: On each documents upload

Maximum: On each documents upload

QUERY PARAMETERS: N/A

FORM PARAMETERS:

Files - [required] The files to be uploaded. Use multipart/form-data with name 'files'

order_documents - OrderDocumentsUploadBean - [required] The list of documents to be uploaded details

JSON:

```
{  
  "order_documents": [  
    {  
      "file_name": String,  
      "type_code": String  
    }]  
}
```

XML:

```
<body>  
  <order_documents>  
    <order_document> [zero or N]  
      <file_name>xsd:string</file_name>  
      <type_code>xsd:string</type_code>  
    </order_document>  
  </order_documents>  
</body>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
------	------	----------	----------	-------------

order_documents	order_document [zero or N]	true	false	
file_name	xsd:string	false	false	The file name
type_code	xsd:string	false	false	The code of the document type

OUTPUT:

OrderDocumentsUploadResultsBean - The errors report. This bean only includes documents with errors. All other documents are successfully uploaded.

JSON:

```
{
  "errors_count": Number,
  "order_documents": [
    {
      "errors": [
        {
          "code": String,
          "field": String,
          "message": String
        }
      ],
      "input": {
        "file_name": String,
        "type_code": String
      }
    }
  ]
}
```

XML:

```
<body>
  <errors_count>xsd:int</errors_count>
  <order_documents>
    <order_document> [zero or N]
      <errors>
        <error> [zero or N]
          <code>xsd:string</code>
          <field>xsd:string</field>
          <message>xsd:string</message>
        </error>
      </errors>
      <input>
        <file_name>xsd:string</file_name>
        <type_code>xsd:string</type_code>
      </input>
    </order_document>
  </order_documents>
</body>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
errors_count	xsd:int	false	false	The number of documents in error state
order_documents	order_document[zero or N]	true	false	Lists of error reports associated to each document in error state
errors	error [zero or N]	true	false	Errors about the input object
code	xsd:string	true	false	Code of the error
field	xsd:string	true	false	Name of the field on the input object. Can be composed, ie: address.zip_code
message	xsd:string	true	false	Message of the error
input	input	false	false	
file_name	xsd:string	false	false	
type_code	xsd:string	false	false	

PRODUCES:

application/json
application/xml

CONSUMES:

multipart/form-data

HTTP RETURN CODES:

200 – Success
404 – if the order can't be found

HTTP Example:

```
POST /api/orders/{order_id}/documents
files=...&order_documents=...
```

P11 - List offers for each product given

METHOD: GET

URL STRUCTURE: /api/products/offers

Note: this resource supports pagination (see documentation). Pagination is carried out on offers, not on products, i.e. maximum total of 'n' offers will be returned if the pagination max param is set to 'n'.

Recommended: On each product page view

Maximum: On each product page view

QUERY PARAMETERS:

product_ids - [required] List of product's skus separated with comma. Limited to 100 values.

offer_state_codes - [required] List of offer state codes

premium - [optional] Boolean : "ALL" (default), "FALSE" or "TRUE". If TRUE (FALSE) returns only offers of premium (not premium) shop.

all_offers - [optional] Boolean : "FALSE" (default) or "TRUE". If FALSE (TRUE) returns only active (all) offers.

channel_codes - [optional] List of the channel codes to filter with, using a comma (,) as a separator.

all_channels - [optional] Boolean: "FALSE" (default) or "TRUE". If FALSE, filter the offers with the given channelCodes or the default channel if no channelCodes are given. If "TRUE", does not filter the offers on channels and ignore the channelCodes.

SORT:

bestPrice - (default) Sort by product sku and then by total price, premium information, shop grade

bestEvaluation - Sort by product sku and then by shop grade, total price, premium information

OUTPUT:

ShopProductsOffersWrapperDto - List of offers for each product given

List of products

JSON:

```
{  
  "products": [  
    {  
      "category_code": String,  
      "category_label": String,  
      "offers": [  
        {  
          "active": Boolean,  
          "available_end_date": Date,  
          "id": String,  
          "offer_state": String,  
          "offer_type": String,  
          "product_id": String,  
          "shop_id": String,  
          "total_price": Number  
        }  
      ]  
    }  
  ]  
}
```

```
"available_start_date": Date,  
"channels": [String],  
"currency_iso_code": "CAD",  
"description": String,  
"discount": {  
    "discount_price": Number,  
    "end_date": Date,  
    "origin_price": Number,  
    "ranges": [{  
        "price": Number,  
        "quantity_threshold": Number  
    }],  
    "start_date": Date  
},  
"favorite_rank": Number,  
"logistic_class": {  
    "code": String,  
    "label": String  
},  
"min_quantity_alert": Number,  
"min_shipping_price": Number,  
"min_shipping_price_additional": Number,  
"min_shipping_type": String,  
"min_shipping_zone": String,  
"nb_evaluation": Number,  
"offer_additional_fields": [{  
    "code": String,  
    "value": String  
}],  
"offer_id": Number,  
"premium": Boolean,  
"price": Number,  
"price_additional_info": String,  
"quantity": Number,  
"shop_grade": Number,  
"shop_name": String,  
"state_code": String,  
"total_price": Number  
},  
"product_references": [{  
    "reference": String,  
    "reference_type": String  
}],  
"product_sku": String,  
"product_title": String,  
"total_count": Number  
}  
}
```

XML:

```
<body>
  <products>
    <product> [zero or N]
      <category_code>xsd:string</category_code>
      <category_label>xsd:string</category_label>
      <offers> [zero or N]
        <active>xsd:boolean</active>
        <available_end_date>xsd:datetime</available_end_date>
        <available_start_date>xsd:datetime</available_start_date>
        <channels>
          <channel_code>xsd:string</channel_code> [zero or N]
        </channels>
        <currency_iso_code>"CAD"</currency_iso_code>
        <description>xsd:string</description>
        <discount>
          <discount_price>xsd:decimal</discount_price>
          <end_date>xsd:datetime</end_date>
          <origin_price>xsd:decimal</origin_price>
          <ranges>
            <range> [zero or N]
              <price>xsd:decimal</price>
              <quantity_threshold>xsd:int</quantity_threshold>
            </range>
          </ranges>
          <start_date>xsd:datetime</start_date>
        </discount>
        <favorite_rank>xsd:int</favorite_rank>
        <logistic_class>
          <code>xsd:string</code>
          <label>xsd:string</label>
        </logistic_class>
        <min_quantity_alert>xsd:int</min_quantity_alert>
        <min_shipping_price>xsd:decimal</min_shipping_price>
        <min_shipping_price_additional>xsd:decimal</min_shipping_price_additional>
        <min_shipping_type>xsd:string</min_shipping_type>
        <min_shipping_zone>xsd:string</min_shipping_zone>
        <nb_evaluation>xsd:long</nb_evaluation>
        <offer_additional_fields>
          <offer_additional_field> [zero or N]
            <code>xsd:string</code>
            <value>xsd:string</value>
          </offer_additional_field>
        </offer_additional_fields>
        <offer_id>xsd:long</offer_id>
        <premium>xsd:boolean</premium>
        <price>xsd:decimal</price>
        <price_additional_info>xsd:string</price_additional_info>
      </product>
    </products>
  </body>
```

```

<quantity>xsd:int</quantity>
<shop_grade>xsd:decimal</shop_grade>
<shop_name>xsd:string</shop_name>
<state_code>xsd:string</state_code>
<total_price>xsd:decimal</total_price>
</offers>
<product_references>
  <product_reference> [zero or N]
    <reference>xsd:string</reference>
    <reference_type>xsd:string</reference_type>
  </product_reference>
</product_references>
<product_sku>xsd:string</product_sku>
<product_title>xsd:string</product_title>
<total_count>xsd:int</total_count>
</product>
</products>
</body>

```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
products	product [zero or N]	false	false	List of products information with their offers
category_code	xsd:string	true	false	Code of the product category
category_label	xsd:string	true	false	Label of the product category
offers	offers [zero or N]	false	false	List of offers for the product
active	xsd:boolean	true	false	
available_end_date	xsd:datetime	false	false	Available end date
available_start_date	xsd:datetime	false	false	Available start date
channels	xsd:string [zero or N]	true	false	List of the channel codes associated to the offer shop
currency_iso_code	"CAD"	true	false	The currency of the shop (iso format)
description	xsd:string	true	true	Description of the offer
discount	discount	false	true	Discount of the offer

discount_price	xsd:decimal	false	true	Discount price. Not to be read if discount with ranges is specified (i.e. discount with at least one range with quantity threshold greater than one).
end_date	xsd:datetime	false	true	Discount end date
origin_price	xsd:decimal	false	true	Origin price
ranges	range [zero or N]	false	false	Discount ranges
price	xsd:decimal	true	false	Price of the offer if the quantity threshold is reached
quantity_threshold	xsd:int	true	false	Minimum quantity to order for the given price to be applied
start_date	xsd:datetime	false	true	Discount start date
favorite_rank	xsd:int	false	false	Favorite rank if the offer is favorite and if the favorite offer feature is enabled
logistic_class	logistic_class	true	true	The logistic class associated to the offer
code	xsd:string	true	false	The code of the logistic class
label	xsd:string	true	false	The label of the logistic class
min_quantity_alert	xsd:int	true	true	Mininum quantity of the offer
min_shipping_price	xsd:decimal	true	false	Unit price of the min shipping rate
min_shipping_price_additional	xsd:decimal	true	false	Additional unit price associated to the min shipping rate
min_shipping_type	xsd:string	true	false	Code of the shipping type associated to the min shipping rate
min_shipping_zone	xsd:string	true	false	Code of the shipping zone associated to the min shipping rate
nb_evaluation	xsd:long	true	false	Number of customers evaluations
offer_additional_fields	offer_additional_field[zero or N]	false	false	List of additional fields
code	xsd:string	true	false	Code of the additional field
value	xsd:string	true	false	Value of the additional field

offer_id	xsd:long	true	false	Identifier of the offer
premium	xsd:boolean	true	false	shop is premium
price	xsd:decimal	true	false	Price of the offer
price_additional_info	xsd:string	false	true	Price additional information of the offer
quantity	xsd:int	true	false	Available quantity of the offer
shop_grade	xsd:decimal	true	true	Grade of the shop which proposes the offer
shop_name	xsd:string	true	false	Name of the shop which proposes the offer
state_code	xsd:string	true	false	State of the offer
total_price	xsd:decimal	true	false	Total price of the offer (price + min shipping rate)
product_references	product_reference[zero or N]	true	false	List of product's references
reference	xsd:string	true	false	Value of the reference
reference_type	xsd:string	true	false	Type of the reference
product_sku	xsd:string	true	false	Sku of the product
product_title	xsd:string	true	false	Title of the product
total_count	xsd:int	true	false	Number of offers for this products

PRODUCES:

application/json

application/xml

HTTP RETURN CODES:

200 – Success

400 – Bad Request if the limit of productIds is exceeded

HTTP Example: GET

```
/api/products/offers?product_ids=...&offer_state_codes=...&premium=...&all_offers=...
&channel_codes=...&all_channels=...
```


P31 - Get products for a list of product's references

METHOD: GET

URL STRUCTURE: /api/products

Note: this resource return 100 products maximum

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS:

product_references - [required] List of the product's identifiants with type (structure : "product_references=<productIdType>|<productId>,<productIdType>|<productId>, ...").
Example: /api/products?product_references=EAN|3120201243238

SORT:

productSku - (default) Sort by product sku and then by product identifier

OUTPUT:

ShopProductsReferencesDto - List of products

JSON:

```
{  
  "products": [  
    {"category_code": String,  
     "category_label": String,  
     "product_id": String,  
     "product_id_type": String,  
     "product_sku": String,  
     "product_title": String  
   ]  
}
```

XML:

```
<body>  
  <products>  
    <product> [zero or N]  
      <category_code>xsd:string</category_code>  
      <category_label>xsd:string</category_label>  
      <product_id>xsd:string</product_id>  
      <product_id_type>xsd:string</product_id_type>  
      <product_sku>xsd:string</product_sku>  
      <product_title>xsd:string</product_title>  
    </product>  
  </products>
```

```
</body>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
products	product [zero or N]	true	false	List of ProductsReferencesBean representation
category_code	xsd:string	true	false	Code of the product category
category_label	xsd:string	true	false	Label of the product category
product_id	xsd:string	true	false	Id of the product
product_id_type	xsd:string	true	false	Id's type of the product
product_sku	xsd:string	true	false	Sku of the product
product_title	xsd:string	true	false	Title of the product

PRODUCES:

application/json

application/xml

HTTP Example: GET /api/products?products=...&product_references=...

RE02 - List of reasons by type

METHOD: GET**URL STRUCTURE:** /api/reasons/{reason_type}**reason_type** - [required] The reason type [INCIDENT_OPEN, INCIDENT_CLOSE, REFUND, OFFER_MESSAGING, ORDER_MESSAGING]**Recommended:** Once on each page view needing a reason**Maximum:** Once on each page view needing a reason**QUERY PARAMETERS:** N/A**SORT:****sortIndex** - (default) Sort by index set in the back-office**OUTPUT:**

ReasonsBean- List of reasons information

List of reasons information

JSON:

```
{
  "reasons": [
    {
      "code": String,
      "label": String
    }
  ]
}
```

XML:

```
<body>
  <reasons>
    <reason> [zero or N]
      <code>xsd:string</code>
      <label>xsd:string</label>
    </reason>
  </reasons>
</body>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
reasons	reason [zero or N]	true	false	List of reasons

code	xsd:string	true	false	Code of the reason
label	xsd:string	true	false	Label of the reason

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/reasons/{reason_type}

S30 - Lists shop's documents

METHOD: GET**URL STRUCTURE:** /api/shops/documents**QUERY PARAMETERS:** N/A**OUTPUT:**

ShopDocumentsBean

JSON:

```
{
  "shop_documents": [
    {
      "date_uploaded": Date,
      "file_name": String,
      "id": Number,
      "type": String
    }
  ]
}
```

XML:

```
<body>
  <shop_documents>
    <shop_document> [zero or N]
      <date_uploaded>xsd:datetime</date_uploaded>
      <file_name>xsd:string</file_name>
      <id>xsd:long</id>
      <type>xsd:string</type>
    </shop_document>
  </shop_documents>
</body>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
shop_documents	shop_document [zero or N]	true	false	The document details
date_uploaded	xsd:datetime	false	false	The upload date of the document
file_name	xsd:string	true	false	The document file name
id	xsd:long	true	false	The identifier of the document

type	xsd:string	true	false	The Shop Document Type code of the document
------	------------	------	-------	---

PRODUCES:

application/json

application/xml

HTTP Example: GET /api/shops/documents

S31 - Downloads one or multiple shop documents

METHOD: GET

URL STRUCTURE: /api/shops/documents/download

Recommended: On each documents download

Maximum: On each documents download

QUERY PARAMETERS:

document_ids - A list of document identifiers (separated by a comma)

type_codes - [optional] A list of document type codes (separated by a comma)

OUTPUT:

Binary

- If a list of document identifiers is specified only these documents are downloaded.
 - If more than one document id is specified, the documents will be wrapped in a ZIP archive
 - If only one document id is specified the document will not be zipped
- If no document identifier is specified, all the shop documents will be downloaded. Use a list of shop document type codes to retrieve specific types from your shop. In this case, the output of the API will always be a ZIP archive even if there is only one document to retrieve.

When documents are retrieved, they're wrapped into a ZIP archive except when only one document id is specified. The tree structure of this archive is as follow:

```
documents-timestamp.zip
| __ shop_id/
|   | __ image.png
|   | __ image(1).png
|   | __ foo.txt
```

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 404 - if at least one document id or one document type code is invalid

HTTP Example: GET /api/shops/documents/download?document_ids=...&type_codes=...

SH21 - List all carriers

METHOD: GET**URL STRUCTURE:** /api/shipping/carriers**Recommended:** 1/day**Maximum:** 1/day**QUERY PARAMETERS:** N/A**SORT:****sortIndex** - (default) Sort by index set in the back-office**OUTPUT:**

CarriersBean - List of carriers

List of carriers without pagination

JSON:

```
{  
  "carriers": [  
    {"code": String,  
     "label": String,  
     "tracking_url": String  
   }]  
}
```

XML:

```
<body>  
  <carriers>  
    <carrier> [zero or N]  
      <code>xsd:string</code>  
      <label>xsd:string</label>  
      <tracking_url>xsd:string</tracking_url>  
    </carrier>  
  </carriers>  
</body>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
carriers	carrier [zero or N]	true	false	List of carriers

code	xsd:string	true	false	Code of the carrier
label	xsd:string	true	false	Label of the carrier
tracking_url	xsd:string	false	true	Tracking url of the carrier

PRODUCES:

application/json

application/xml

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/shipping/carriers

Catalog/Products

H11 - List hierarchies related (parents and children) to the given hierarchy

METHOD: GET

URL STRUCTURE: /api/hierarchies

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS:

hierarchy - [optional] The code of the hierarchy

max_level - [optional] Number of children hierarchy levels to retrieve. If not specified, all child hierarchies are retrieved

OUTPUT:

HierarchiesBean - The list of hierarchies

List of hierarchies

JSON:

```
{  
  "hierarchies": [ {  
    "code": String,  
    "label": String,  
    "level": Number,  
    "parent_code": String  
  } ]  
}
```

XML:

```
<body>  
  <hierarchies>  
    <hierarchy> [zero or N]  
      <code>xsd:string</code>  
      <label>xsd:string</label>  
      <level>xsd:int</level>  
      <parent_code>xsd:string</parent_code>  
    </hierarchy>  
  </hierarchies>  
</body>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
hierarchies	hierarchy [zero or N]	true	false	List of hierarchies
code	xsd:string	true	false	Hierarchy code
label	xsd:string	true	false	Hierarchy label
level	xsd:int	true	false	Hierarchy level
parent_code	xsd:string	false	true	Parent Hierarchy code

PRODUCES:

application/xml

application/json

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/hierarchies?hierarchy=...&max_level=...

P41 - Import offers: import file to add products

METHOD: POST

URL STRUCTURE: /api/products/imports

File should be uploaded.

Returns the import identifier to track the status of the import.

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS: N/A

FORM PARAMETERS:

file - [required] Import file to upload. Use multipart/form-data with name 'file'.

See [Appendix B](#) for csv format.

OUTPUT:

ProductImportTrackingBean - Identifier of the import

Product import tracking information

JSON:

```
{  
  "import_id": Number  
}
```

XML:

```
<product_import_tracking>  
  <import_id>xsd:long</import_id>  
</product_import_tracking>
```

Attribute details:

Elements

Name	type	Required	Nillable	Description
import_id	xsd:long	true	false	Identifier of the import

PRODUCES:

application/json

application/xml

CONSUMES:

multipart/form-data

HTTP RETURN CODES: 201 - Created

HTTP RESPONSE HEADERS:

Location - URI of the new import.

HTTP Example: POST /api/products/imports

P42 - Get import status

METHOD: GET**URL STRUCTURE:** /api/products/imports/{import}**import** - [required] The identifier of the import**Recommended:** 12/hour**Maximum:** 60/hour**QUERY PARAMETERS:** N/A**OUTPUT:**

ProductImportStatusTrackingBean - Information of the import

JSON:

```
{  
    "date_created": Date,  
    "has_error_report": Boolean,  
    "has_new_product_report": Boolean,  
    "has_transformation_error_report": Boolean,  
    "has_transformed_file": Boolean,  
    "import_id": Number,  
    "import_status": String,  
    "reason_status": String,  
    "transform_lines_in_error": Number,  
    "transform_lines_in_succes": Number,  
    "transform_lines_read": Number,  
    "transform_lines_with_warning": Number  
}
```

XML:

```
<product_import_tracking>  
    <date_created>xsd:datetime</date_created>  
    <has_error_report>xsd:boolean</has_error_report>  
    <has_new_product_report>xsd:boolean</has_new_product_report>  
    <has_transformation_error_report>xsd:boolean</has_transformation_error_report>  
    <has_transformed_file>xsd:boolean</has_transformed_file>  
    <import_id>xsd:long</import_id>  
    <import_status>xsd:string</import_status>  
    <reason_status>xsd:string</reason_status>  
    <transform_lines_in_error>xsd:long</transform_lines_in_error>  
    <transform_lines_in_succes>xsd:long</transform_lines_in_succes>  
    <transform_lines_read>xsd:long</transform_lines_read>  
    <transform_lines_with_warning>xsd:long</transform_lines_with_warning>  
</product_import_tracking>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
date_created	xsd:datetime	true	false	Creation date of the import
has_error_report	xsd:boolean	true	false	Returns true if error report is available. Value is filled when the import is completed
has_new_product_report	xsd:boolean	true	false	Returns true if new product report is available. Value is filled when the import is completed
has_transformation_error_report	xsd:boolean	false	false	Returns true if transformation error report is available. Value is filled when the import is completed
has_transformed_file	xsd:boolean	true	false	Returns true if transformed file is available. Value is filled when the import is completed
import_id	xsd:long	true	false	Identifier of the import
import_status	xsd:string	true	false	Identifier of the import
reason_status	xsd:string	false	false	A message explaining the reason of the import status, if relevant
transform_lines_in_error	xsd:long	true	false	Total count of transformed lines in error
transform_lines_in_succes	xsd:long	true	false	Total count of transformed lines in success
transform_lines_read	xsd:long	true	false	Total count of transformed lines read
transform_lines_with_warning	xsd:long	true	false	Total count of transformed lines with warning

PRODUCES:

application/json
application/xml

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/products/imports/{import}

P44 - Get errors report file for an import**METHOD:** GET**URL STRUCTURE:** /api/products/imports/{import}/error_report**import** - [required] The identifier of the import**Recommended:** 12/hour**Maximum:** 60/hour**QUERY PARAMETERS:** N/A**OUTPUT:**

Binary - The error report file

HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/products/imports/{import}/error_report

P45 - Get products report file for an import

METHOD: GET

URL STRUCTURE: /api/products/imports/{import}/new_product_report

import - [required] The identifier of the import

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS: N/A

OUTPUT:

Binary - The products report file

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/products/imports/{import}/new_product_report

P46 - Get transformed file for an import

METHOD: GET

URL STRUCTURE: /api/products/imports/{import}/transformed_file

import - [required] The identifier of the import

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS: N/A

OUTPUT:

Binary - The transformed file

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/products/imports/{import}/transformed_file

P47 - Get transformation errors report file for an import

METHOD: GET

URL STRUCTURE: /api/products/imports/{import}/transformation_error_report

import - [required] The identifier of the import

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS: N/A

OUTPUT:

Binary - The transformation error report file

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/products/imports/{import}/transformation_error_report

PM11 - Get attributes configuration

METHOD: GET

URL STRUCTURE: /api/products/attributes

Retrieves all attributes for parents and children of the requested hierarchy

Recommended: 12/hour

Maximum: 60/hour

QUERY PARAMETERS:

hierarchy - [optional] Code of the hierarchy. If not specified, all attributes are retrieved

max_level - [optional] Number of children hierarchy levels to retrieve. If not specified, attributes of all child hierarchies are retrieved

OUTPUT:

AttributesConfigurationBean - List of attributes configuration

List of attribute configuration

JSON:

```
{  
  "attributes": [{  
    "code": String,  
    "default_value": String,  
    "description": String,  
    "example": String,  
    "hierarchy_code": String,  
    "label": String,  
    "required": Boolean,  
    "type": String,  
    "type_parameter": String,  
    "variant": Boolean  
  }]  
}
```

XML:

```
<body>  
  <attributes>  
    <attribute> [zero or N]  
      <code>xsd:string</code>  
      <default_value>xsd:string</default_value>  
      <description>xsd:string</description>  
      <example>xsd:string</example>  
      <hierarchy_code>xsd:string</hierarchy_code>  
      <label>xsd:string</label>  
      <required>xsd:boolean</required>
```

```

<type>xsd:string</type>
<type_parameter>xsd:string</type_parameter>
<variant>xsd:boolean</variant>
</attribute>
</attributes>
</body>

```

Attribute details:

Elements

Name	type	Required	Nillable	Description
attributes	attribute [zero or N]	true	false	List of attribute configuration
code	xsd:string	true	false	Code
default_value	xsd:string	false	true	Value from the list linked to the attribute
description	xsd:string	false	true	Description
example	xsd:string	false	true	Example
hierarchy_code	xsd:string	false	true	The hierarchy this attribute belongs to. If this code is empty, the attribute is shared by all hierarchies.
label	xsd:string	false	true	Label
required	xsd:boolean	true	false	Attribute is required
type	xsd:string	true	false	Type
type_parameter	xsd:string	true	true	Type parameter
variant	xsd:boolean	true	false	Attribute is variant

PRODUCES:

application/xml

application/json

HTTP RETURN CODES: 200 – Success

HTTP Example: GET /api/products/attributes?hierarchy=...&max_level=...

PM31 - Returns values for given operator's values list

METHOD: GET

URL STRUCTURE: /api/products/attributes/values_list/{values_list}

values_list(.+) - [required] The operator's value list code

Recommended: 16/day

Maximum: 32/day

QUERY PARAMETERS: N/A

OUTPUT:

ValuesListBean - Operator Values List representation

A list of values to restrict an attribute possible values.

JSON:

```
{  
    "code": String,  
    "label": String,  
    "values": [  
        {"code": String,  
         "label": String  
    }]  
}
```

XML:

```
<values_list>  
    <code>xsd:string</code>  
    <label>xsd:string</label>  
    <values>  
        <value> [zero or N]  
            <code>xsd:string</code>  
            <label>xsd:string</label>  
        </value>  
    </values>  
</values_list>
```

Attribute details:

Elements

Name	Type	Required	Nillable	Description
code	xsd:string	true	false	The list's identifier

label	xsd:string	false	true	The list's label
values	value [zero or N]	false	false	The list's values
code	xsd:string	true	false	The value identifier
label	xsd:string	false	true	The value label

PRODUCES:

application/json

application/xml

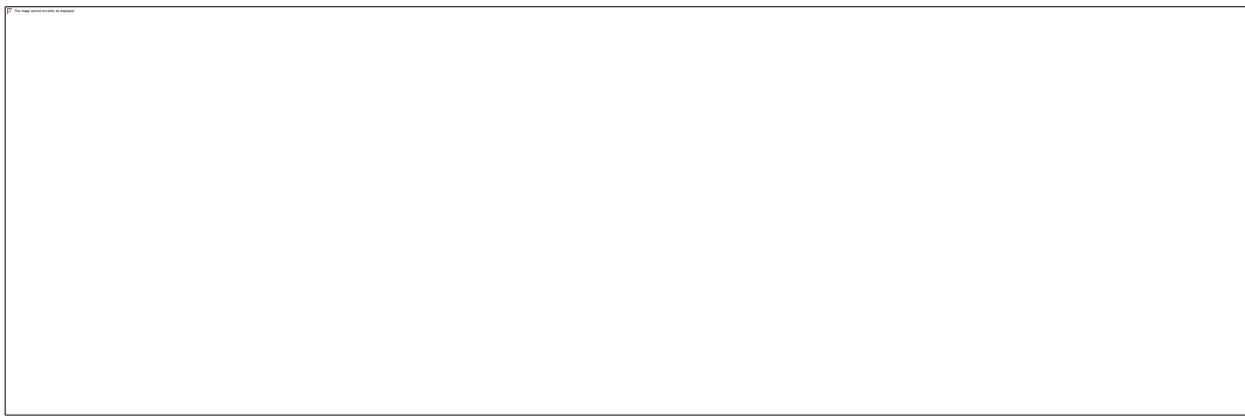
HTTP RETURN CODES: 200 – Success**HTTP Example:** GET /api/products/attributes/values_list/{values_list}

Appendecies

Appendix A – Retrieve your shop API key

Login to your account at

<https://marketplace.bestbuy.ca/>

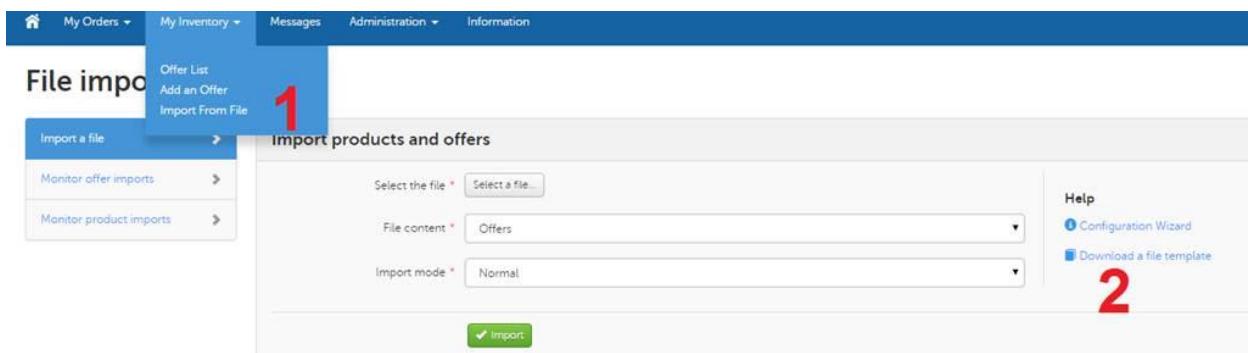


1. Click on your login name to access your account settings
2. Click on "API Key" menu
3. Click on the "Generate a new key" button to get a Shop Key
4. Contact marketplacesupport@bestbuycanada.ca for the Operator Front Key.

Appendix B – Download form templates for OF01 and P41

The template is dependent on the category which the products will be loaded into. The templates for the categories are available on the portal whether you're loading through API or Portal.

Login to your account at
<https://marketplace.bestbuy.ca/>



1. From the "My Inventory" menu, select "Import From File".
2. From the "File content" drop-down, select either:
 - a) "Offers" for OF01 Offers Only
 - b) "Products and Offers" for OF01 Offers With Products
 - c) "Products" for P41 Products only
3. Click on "Download a file template"



4. Select the template for the appropriate category.

Note: Please select CSV vs XML as the API can only use CSV.

Remember that [OF01](#) or [P41](#) can be used to load products.

The advantage to using [OF01](#) for "Products and Offers" is that you don't have to wait for the product to load before creating an offer.

The disadvantage to using [OF01](#) is that you remove control over when the product will go live on the website. Go live happens as soon as it's ready to be processed overnight.

Appendix C – Postman Compatible JSON Collection

Install [Postman](#) as a [Google Chrome](#) extension

Contact marketplacesupport@bestbuycanada.ca for a Postman compatible JSON Collection for test purposes.