Myntra Seller API

Version: 2.0

Changes from v1:

- Order APIs take sku instead of orderLineId.
- Added confirmOrder API
- Added GST based changes in updateTax and ReadyToDispatch
- Corrected missing a braces for status response.
- Changed Return status names.
- Added summary of success and error codes at the bottom.
- Added Price Update API
- Added Catalog Attribute Sheet Status API - Only for sellers using FTP for sharing attribute sheet upload.
- Changed the URL format for the APIs. Now version comes after the "seller" part - ("seller/v2")

- Note:
- Inbound Calls
  - Authentication
    - Obtain Auth token
  - Catalog Attribute Sheet Status
    - Lot Details
  - Price Update
    - Price Update
    - Job Monitoring
  - Discounts
    - Update Discount
  - Inventory
    - Update Inventory
    - Search Inventory
  - Order Updates
    - Confirm Order
    - Update Tax
    - Ready To Dispatch (RTD)
    - Cancel Order Line
  - Order Search and Document Download
    - Get Order by Id
    - Download Shipping Label
  - Logistics
    - Verify Shipments
    - Get Manifest
    - Tracking Updates
- Outbound Calls
  - Order
    - Create Order
    - Update Order
    - Download Invoice
  - Return
    - Return state transition diagram
    - Create Return
    - Update Return
    - Return Delivered
    - Return Cancelled
    - Return Ready For Pickup

Note:

- You will have to ignore attributes other than those defined below that you receive in the request/response.
- The "sku" attribute in the APIs is the seller sku code.
- The timezone of date attributes are IST by default.
- Status codes are NOT Http Codes; Http Response code is 200 by default unless explicitly specified.
- Currently Shipments correspond one-to-one with Orders and have the same identifier. i.e. you may pass orderId as shipmentId.

Inbound Calls

These are the set of APIs which sellers can invoke on Myntra marketplace system

Authentication

Obtain Auth token

Endpoint	https://partnerapi.myntra.com/v1/authenticate
Http Method	POST
HttpHeader	base64 encoded <username>~<username>:<password> in the Basic Authorisation
Description	This API should be used by the seller to submit the username and password to receive the auth token, which is required in the header to invoke all the other seller APIs. The token may be used for at most 30 days after which the API will throw a 401. It is advised to code in such a way that when a 401 is thrown for any API, this API is called to retrieve the latest token.
Sample Http Header	Authorization: Basic dkkdk323dssssdddffr322333333223==
Response	

Param	DataType	Description
authToken	String	The auth token which is a required header for all further API calls.
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	ENUM	SUCCESS, ERROR

Sample Response


```
{
  "status": {
    "statusCode": 1000,
    "statusMessage": "Token retrieved successfully",
    "statusType": "SUCCESS"
  },
  "data": {
    "authToken": "dajsdjhfkjdhskjahfdkjahsdfkjhakfhdksajhfalsdkfkjdshfkjdhadkjfds"
  }
}
```


Status Codes

Status Code	Status Type	Case
1000	SUCCESS	Success
401	ERROR	Unauthorised

Catalog Attribute Sheet Status

Lot Details

Endpoint	https://partnerapi.myntra.com/seller/v2/catalog/lot/<lotname>
Details	Get details of a LOT by lot name submitted through partner APIs
Http Method	GET
HttpHeader	Authorization: Bearer <Access-Token>
Description	To get the details for LOT by LOT name
Path Param	lot name
Request Params	None
Response Params	

Param	Data type	Description
StatusCode	int	1000 for success and 2000 for ERROR
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.
LOTDetails	Json Object	Details of the LOT

Sample Request	https://partnerapi.myntra.com/seller/v2/catalog/lot/test

LOT states	`UPLOADED,`
	`IMAGE_VALIDATED,`
	`MANUAL_QC,`
	`LOT_PARTIALLY_ACCEPTED,`
	`LOT_REJECTED,`
	`LOT_ACCEPTED,`
	`LOT_STYLES_CREATED,`
	`LOT_STYLES_LIVE`

Sample Response


```
{
  "status": {
    "statusCode": 1000,
    "statusMessage": "Success",
    "statusType": "SUCCESS",
    "totalCount": 1
  },
  "data": [
    {
      "id": 1331,
      "createdBy": "system",
      "createdOn": 1491218963000,
      "lastModifiedOn": 1491939261000,
      "version": 2,
      "status": "Uploaded",
      "sellerId": 2655,
      "inputFileLocation":
"http://myntra.myntassets.com/assets/documents/2017/4/3/81595-1491218962
      "studio": "other_studios",
      "businessUnit": "Women's Ethnic",
      "brandName": "Pitney Bowes",
      "productsCount": 91,
   "lotname":"test",
   "source":"seller"
    }
  ]
}
```


Endpoint	https://partnerapi.myntra.com/seller/v2/catalog/lot/<lotid>
Details	Get details of a LOT by lot ID
Http Method	GET
HttpHeader	Authorization: Bearer <Access-Token>
Description	To get the details for LOT by LOTId
Path Param	lot name
Request Params	None
Response Params	

Param	Data type	Description
StatusCode	int	200 for success and 500 for ERROR
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.
LOTDetails	Json Object	Details of the LOT

Sample Request	https://partnerapi.myntra.com/seller/v2/catalog/lot/1331
LOT states	UPLOADED, IMAGE_VALIDATED, MANUAL_QC, LOT_PARTIALLY_ACCEPTED, LOT_REJECTED, LOT_ACCEPTED, LOT_STYLES_CREATED, LOT_STYLES_LIVE

Sample Response	{
	"status": {
	"statusCode": 1000,
	"statusMessage": "Success",
	"statusType": "SUCCESS",
	"totalCount": 1
	},
	"data": [
	{
	"id": 1331,
	"createdBy": "system",
	"createdOn": 1491218963000,
	"lastModifiedOn": 1491939261000,
	"version": 2,
	"status": "Uploaded",
	"sellerId": 2655,
	"inputFileLocation":
	"http://myntra.myntassets.com/assets/documents/2017/4/3/81595-1491218962
	"studio": "other_studios",
	"businessUnit": "Women's Ethnic",
	"brandName": "Pitney Bowes",
	"productsCount": 91,
	"lotname":"test",
	"source":"seller"
	}
	]
	}

Price Update

Price Update

Endpoint	https://partnerapi.myntra.com/seller/v2/catalog/updatePrice
Details	Update Mrp and Selling Price for styles listed on Myntra • Takes in seller SKU code as input and updates both Mrp and selling Price of the style under which a SKU belongs • If two SKUs of the same style is part of one request, the last encountered SKU's Mrp and Selling Price will be set for the style
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>

Description	To update MRP of listed products in Myntra
Body	JSON Payload (Refer sample request)
Constraints	Max 2000 seller SKUs can be uploaded at one go
Request Params	None
Response Params	

Param	DataType	Description
StatusCode	int	1000 for success and 2000 for ERROR
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.
JobID	Int	Reference to the created job

Sample Request	

```
{
   "skuList":[
      {
         "sellerSKU":"SKU1",
         "mrp":{
            "total":1234.50,
            "shipping_charges":200.50,
            "taxes_and_duties":34.50,
            "currency":"INR"
         },
         "sellingPrice":{
            "total":1234.50,
            "shipping_charges":200.50,
            "taxes_and_duties":34.50,
            "currency":"INR"
         }
      },
      {
         "sellerSKU":"SKU2",
         "mrp":{
            "total":2234.50,
            "shipping_charges":200.50,
            "taxes_and_duties":34.50,
            "currency":"INR"
         },
         "sellingPrice":{
            "total":2234.50,
            "shipping_charges":200.50,
            "taxes_and_duties":34.50,
            "currency":"INR"
         }
      }
   ],
   "comment":"REASON_FOR_CHANGE"
}
```

Sample Response	{
	"status":{
	"statusCode":200,
	"statusMessage":"Success",
	"statusType":"SUCCESS"
	},
	"data":{
	"jobId":26917,
	"createdBy":"seller.api",
	"createdOn":1484742593000,
	"lastModifiedOn":1484744642000,
	"version":1,
	"startTime":1484742593000,
	"totalRecords":2,
	"processedRecords":0,
	"erredRecords":0,
	"numberOfChunks":14,
	"modifiedBy":"seller.api",
	"status":"SUBMITTED",
	"metaData":"{\"comment\":\"REASON_FOR_CHANGE\"}"
	}
	}

Job Monitoring

Endpoint	https://partnerapi.myntra.com/seller/v2/catalog/job/<jobId>
Details	Get details of a Job submitted through partner APIs
Http Method	GET
HttpHeader	Authorization: Bearer <Access-Token>
Description	To update MRP of listed products in Myntra
Path Param	Job ID you want to monitor
Request Params	None
Response Params	

Param	DataType	Description
StatusCode	int	1000 for success and 2000 for ERROR
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.
JobDetails	Json Object	Details of the job

Sample Request	https://partnerapi.myntra.com/seller/v2/catalog/job/26917

Job States	`Processing | Completed | Erred`
Note	Details of both updated and failed records will be part of the metaResults block which will be a JSON encoded string. This can be p
Sample Response	


```
{
 "status": {
  "statusCode": 3,
  "statusMessage": "Success",
  "statusType": "SUCCESS",
  "totalCount": 1
 },
 "data": [{
  "id": 26917,
  "createdBy":"seller.api",
  "createdOn":1484742593000,
  "lastModifiedOn":1484744642000,
  "version":1,
  "startTime":1484742593000,
  "endTime":1489152953651,
  "totalRecords":2,
  "processedRecords":1,
  "erredRecords":1,
  "numberOfChunks":2,
  "modifiedBy":"seller.api",
  "status":"Completed",
  "subJobsEntryList":[{
   "id":1076128,
   "status":"Completed",
   "erredRecords":0,
   "processedRecords":1,
   "totalRecords":1,
   "parentJobId":26917,
   "metaResults":"{\"junkData\":[],\"errors\":[],\"errorRecords\":[],\"p
  },
  {
   "id":1076129,
   "status":"Completed",
   "modifiedBy":"CMS Migration User",
   "erredRecords":1,
   "processedRecords":0,
   "totalRecords":1,
   "parentJobId":26917,
   "metaResults":"{\"junkData\":[],\"errors\":{\"1\":[
{\"errorMessage\":\"Incorrect SKU id SKU2\",\"recordId\":\"SKU2\"}]},
\"errorRecords\":[\"SKU2\"],\"processedRecords\":[]}"
  }],
  "metaData":"{\"comment\":\"REASON_FOR_CHANGE\"}"
 }]
}
```

Discounts

Update Discount

Endpoint	https://partnerapi.myntra.com/seller/v2/discount/update
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>
Description	API to be used to perform discount updates to updatemyntrawith the discount for a particular seller SKU code, for agivedate range. If multiple SKUs in the request belonging to the same style are given different discount values, then the least of the discounts will be applied for the entire style. The above discount will overwrite any existing discount for any of the SKUs belonging to the same style for the date range provided in the request. Maximum number of SKUs allowed in one call is 10. DiscountEnd date has to < 180 days from the start date.Formatof state date and end date is java date. Time can be provided as well, please check the sample request. In order to remove the discount, discount needs to be set to 0, startDate - the date time effective which the discount needs to be removed. End date is empty.
Request Params	List[Discount]

Param	DataType	Required	Description	Field Length
discount	decimal	Y	The percentage trade discount provided by the seller	12,2
sku	string	Y	The seller's sku for the product	128
startDate	Date	Y	The start date for the discount	Format : "dd-MM-yyyy HH:mm:ss"
endDate	Date	Y	The end date for the discount	Format : "dd-MM-yyyy HH:mm:ss"

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Sample Request

```
[
  {
    "discount": 10,
    "sku": "ABCDEE2322",
    "startDate": "22-06-2016 10:20:56",
    "endDate": "22-07-2016 10:20:56"
  },
  {
    "discount": 10,
    "sku": "SKU1232",
    "startDate": "22-06-2016 10:20:56",
    "endDate": "22-07-2016 10:20:56"
  }
]
```

Sample Success Response	{
	"status": {
	"statusCode": 1006,
	"statusMessage": "Discounts processed successfully",
	"statusType": "SUCCESS",
	"totalCount": 0
	}
	}

Sample Partial Success Response	{
	"status": {
	"statusCode": 1006,
	"statusMessage": "Discounts processed successfully",
	"statusType": "SUCCESS",
	"totalCount": 0
	},
	"discountEntries": [
	{
	"sku": "SKU1232",
	"remarks": "Invalid SKU id"
	}
	]
	}

Status Codes	

Status Code	Status Type	Case
1006	SUCCESS	Discount processed successfully
2003	ERROR	Error while processing discount

Inventory

Update Inventory

Endpoint	https://partnerapi.myntra.com/seller/v2/inventory/update
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>
Description	API to be used to perform inventory updates to update myntra with the latest inventory snapshot and the lead time for seller sku

Request Params

Param	DataType	Required	Description	Field Length
inventoryCount	int	Y	The actual snapshot of inventory present in the seller warehouse at any point of time	11
sku	string	Y	The seller's sku for the product	128
processingSla	int	Y	The time required by the seller to have the item ready for dispatch (in number of days)	11
warehouse	string	Y	The seller warehouse code at which the SKU is available	50

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Sample Request

```
[
  {
    "inventoryCount": 10,
    "sku": "ABCDEE2322",
    "processingSla": 5,
    "warehouse": "MF"
  },
  {
    "inventoryCount": 10,
    "sku": "ABCDEE2323",
    "processingSla": 5,
    "warehouse": "MF"
  }
]
```

Sample Success Response

```
{
  "status": {
    "statusCode": 1001,
    "statusMessage": "Inventory processed successfully",
    "statusType": "SUCCESS"
  }
}
```

Sample Partial Success Response	{
	"status": {
	"statusCode": 1001,
	"statusMessage": "Inventory processed successfully",
	"statusType": "SUCCESS",
	"totalCount": 0
	},
	"inventoryEntries": [
	{
	"inventoryCount": 10,
	"sku": "ABCDEE2323",
	"processingSla": 5,
	"warehouse": "MF",
	"remarks": "Invalid SKU"
	}
	]
	}

Status Codes	

Status Code	Status Type	Case
1001	SUCCESS	Inventory updated successfully
2001	ERROR	Error while updating inventory

Search Inventory

Endpoint	https://partnerapi.myntra.com/seller/v2/inventory/search
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>
Description	API to be used to search inventory by seller sku codes.
Request Params	

Param	DataType	Required	Description	Field Length
List<sku>	List	Y	The list of seller SKU code for which inventory is to be retrieved, in batches of 20.	128 (This length is of individual sku)

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR
List<Inventory>		List of inventory objects, whose properties are described below
inventoryCount	int	The actual snapshot of inventory present in the seller warehouse at any point of time
sku	string	The seller's sku for the product
processingSla	int	The time required by the seller to have the sku code ready for dispatch (in days)
warehouse	string	The seller's warehouse code

Sample Request	["8903880486532"]

Sample Response	{
	"status": {
	"statusCode": 1002,
	"statusMessage": "Inventory retrieved successfully",
	"statusType": "SUCCESS",
	"totalCount": 0
	},
	"inventoryEntries": [
	{
	"inventoryCount": 95,
	"sku": "8903880486532",
	"warehouse": "4"
	}
	]
	}

Status Codes	

Status Code	Status Type	Case
1002	SUCCESS	Inventory retrieved successfully
2002	ERROR	Error while retrieving inventory

Order Updates

Confirm Order

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{orderId}/confirmOrder
Http Method	PUT
HttpHeader	Authorization: Bearer <Access-Token>
Description	Order confirmation API. Myntra will not allow customer cancellation on PB orders after this call received successfullly
Path Param	orderId
Response	200 OK
Request Params	

RequestParam	DataType	Required	Description	Field Length
List<OrderLine>				
sku	string	Y	The seller SKU code for which order was placed.	128
quantity	int	Y	The number of items that are being packed.	11

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Request

```
[
  {
    "sku": "SKU1",
    "quantity": 1
  }
]
```

Sample Response

```
{
  "status": {
    "statusCode": 1003,
    "statusMessage": "Order updated successfully",
    "statusType": "SUCCESS"
  }
}
```

Status Codes

Status Code	Status Type	Case
1003	SUCCESS	Order updated successfully
2033	ERROR	Order is already confirmed
2006	ERROR	Invalid payload data - No Payload found, Order ID Not Present in payload, Order Release Lines not found in payload
2034	ERROR	Seller ID for order line ID is different
2020	ERROR	Order ID not found in DB, Order Lines is not found for given order ID
2031	ERROR	Extra order lines in the payload, Order lines are missing in the payload
2032	ERROR	Quantity mismatch with any order line
2004	ERROR	Business Validation failure

Validations in API

- Validate request

- order id, order lines,skuidnot null

- Seller id at each line is present

- All SKU in request should actually be part of the order

- Validate Order status
- order exists
- order should be in confirmed status

Update Tax

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{orderId}/updateTax/
Http Method	PUT
HttpHeader	Authorization: Bearer <Access-Token>
Description	To update the taxes in Myntra systems. The ready to dispatch call will fail if this API was not invoked.
Path Param	orderId

Request Params

RequestParam	DataType	Required	Description	Field Length
List<OrderLine>				
sku	string	Y	The seller SKU code for which order was placed.	128
invoiceNumber	string	Y	Invoice number	
invoiceDate	Date	Y	Invoice date	Format : "dd-MM-yyyy HH:mm:ss"
List<TaxEntries>				
unitTaxAmount	double	Y	The tax amount for a single unit of this order line	12,2
taxRate	double	Y	The rate of tax	12,2
taxType	string	Y	The type of tax VAT/CST/CGST/SGST/IGST/CESS	128
unitTaxableAmount	double	Y	Total taxable amount per item	12,2

Sample Request

```
[
  {
    "sku": "8903988586813",
 "invoiceNumber": "12345",
    "invoiceDate": "10-01-2017 10:20:10",
    "taxEntries": [
      {
        "taxRate": 5.5,
        "taxType": "CGST",
        "unitTaxAmount": 26.02,
        "unitTaxableAmount": 711.45
      },
      {
        "taxRate": 5.5,
        "taxType": "SGST",
        "unitTaxAmount": 26.02,
        "unitTaxableAmount": 711.45
      }
    ]
  }
]
```

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Response	{
	"status": {
	"statusCode": 1003,
	"statusMessage": "Order updated successfully",
	"statusType": "SUCCESS"
	}
	}

Status Codes	<table><tr><th>Status Code</th><th>Status Type</th><th>Case</th></tr><tr><td>1003</td><td>SUCCESS</td><td>Tax updated successfully</td></tr><tr><td>2006</td><td>ERROR</td><td>Invalid payload data</td></tr><tr><td>2004</td><td>ERROR</td><td>Business Validation failure</td></tr></table>
Validations in API	▪ Validate request - order id, order lines, sku, invoice number, invoice date not null - Seller id at each line is present - All SKU in request should actually be part of the order ▪ Validate tax info - tax entry, tax type, tax rate, tax amount, unit tax amount not null - Tax Type should be one of the valid list explained above - If VAT/CST is applied, then GST should not be applied. - In VAT regime, only one of VAT or CST can be applied at a time. - In GST regime, CGST and SGST must be sent together and IGST must be sent alone (as applicable)

Ready To Dispatch (RTD)

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{orderId}/readyToDispatch/
Http Method	PUT
HttpHeader	Authorization: Bearer <Access-Token>
Description	To mark the order ready to dispatch in Myntra. This call will fail if update tax API was not invoked.
Path Param	orderId

Request Params

RequestParam	DataType	Required	Description	Field Length
List<OrderLine>				
sku	string	Y	The seller SKU code for which order was placed.	128
quantity	int	Y	The number of items that are being packed.	11
invoiceNumber	string	Y	Invoice number	
invoiceDate	Date	Y	Invoice date	Format : "dd-MM-yyyy HH:mm:ss"
List<TaxEntries>			Not required in case of International sellers	
unitTaxAmount	double	Y	The tax amount for a single unit of this order line	12,2
taxRate	double	Y	The rate of tax	12,2
taxType	string	Y	The type of tax VAT/CST/CGST/SGST/IGST/CESS	128
unitTaxableAmount	double	Y	Total taxable amount per item	12,2

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Request


```
[
  {
    "sku": "SKU1",
    "quantity": 1,
    "invoiceNumber": "12345",
    "invoiceDate": "10-01-2017 10:20:10",
    "taxEntries": [
      {
        "taxRate": 5.5,
        "taxType": "CGST",
        "unitTaxAmount": 26.02,
        "unitTaxableAmount": 711.45
      },
      {
        "taxRate": 5.5,
        "taxType": "SGST",
        "unitTaxAmount": 26.02,
        "unitTaxableAmount": 711.45
      }
    ]
  }
]
```

Sample Response	{
	"status": {
	"statusCode": 1003,
	"statusMessage": "Order updated successfully",
	"statusType": "SUCCESS"
	}
	}

Status Codes	

Status Code	Status Type	Case
1003	SUCCESS	Order updated successfully
2005	ERROR	Duplicate Request
2006	ERROR	Invalid payload data
2004	ERROR	Business Validation failure

Validations in the API	- Validate Request - order id, order lines, SKU not null - Seller id present at all order lines - Validate Order status - order exists - order should not be on hold - order should be in confirmed status - courier code and tracking number should be present - Packed date should not be before seller processing start time - All sku in request should be a part of release - Tax rate and unit tax amount should have been update for all order lines (via the updateTax API) - Check whether the quantity being packed for each order line matches the full order line quantity.

Cancel Order Line

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{orderId}/cancelItems
Http Method	PUT
HttpHeader	Authorization: Bearer <Access-Token>
Description	To cancel the order lines in myntra marketplace. The order will be marked as cancelled if all the lines are cancelled and cancellation type will be marked 'Seller Failed To Supply'
Path Param	orderId
Request Params	

RequestParam	DataType	Required	Description	Field Length
List<OrderLine>				
sku	string	Y	The seller SKU code for which order was placed.	128
quantity	int	Y	The absolute count of cancelled items.	11

Response Params	

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Request


```
[
  {
    "sku": "SKU1",
    "quantity": 1
  }
]
```


Sample Response


```
{
  "status": {
    "statusCode": 1004,
    "statusMessage": "Items cancelled successfully",
    "statusType": "SUCCESS"
  }
}
```


Status Codes

Status Code	Status Type	Case
1004	SUCCESS	Items cancelled successfully
2005	ERROR	Duplicate Request
2006	ERROR	Invalid payload data
2004	ERROR	Business Validation failure

Validations in API

- Cancellation reason is present and is valid
- Seller id present at all order lines
- All SKUs in request should be a part of release
- Order should be in confirmed but not packed, shipped, or delivered.
- No. of lines to be cancelled should be non-empty
- Quantity to be cancelled should be <= total un-cancelled quantity of the order line.

Order Search and Document Download

Get Order by Id

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{id}
Http Method	GET
HttpHeader	Authorization: Bearer <Access-Token>
Description	To search for an order by its id
Path Param	orderId

Response Params	The order object.
Sample Request	https://partnerapi.myntra.com/seller/v2/order/12322323

Sample Response	{
	"status": {
	"statusCode": 1005,
	"statusMessage": "Order retrieved successfully",
	"statusType": "SUCCESS",
	"totalCount": 0
	},
	"orderEntry": {
	"id": 2147460075,
	"orderLineEntries": [
	{
	"id": 895234,
	"orderId": 2147460075,
	"totalQuantity": 1,
	"mrp": 799,
	"totalSellerDiscount": 0,
	"totalMarketplaceDiscount": 0,
	"totalPaymentReceived": 0,
	"unitTaxAmount": 0
	}
	],
	"shippingCharge": 0,
	"giftCharge": 0,
	"amountDue": 799,
	"receiverName": "Abhijit Pati",
	"address": "Myntra Design Pvt. Ltd.",
	"city": "Bangalore",
	"locality": "Begur",
	"state": "KA",
	"zipcode": "560068",
	"country": "India",
	"mobile": "2121212121",
	"warehouse": "1",
	"onHold": false,
	"orderDate": "28-02-2017 20:37:32",

"shippingMethod": "NORMAL",
"expectedDeliveryTime": "04-03-2017 21:00:01",
"processingStartTime": "01-03-2017 16:50:01",
"processingCutoffTime": "01-03-2017 21:00:01",
"courierCode": "ML",
"paymentMethod": "cod"
}
}

Status Codes

Status Code	Status Type	Case
1005	SUCCESS	Order retrieved successfully
2020	ERROR	Error while retrieving order

Download Shipping Label

Endpoint	https://partnerapi.myntra.com/seller/v2/order/{orderId}/shippingLabel/
Http Method	GET
HttpHeader	Authorization: Bearer <Access-Token>
Description	To download the shipping label for a given orderId
Path Param	orderId
Response Params	

Param	DataType	Description
ShippingLabelByteStream	`application/pdf`	The shipping label pdf in a byte stream format

Sample Request	https://partnerapi.myntra.com/seller/v2/order/2147553448/shippingLabel/

Logistics

Verify Shipments

Endpoint	https://partnerapi.myntra.com/seller/v2/logistics/verifyReadyToShipShipments
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>
Description	API to be used to verify shipments against a courier code and corresponding order ids.
Request Params	

Param	DataType	Required	Description	Field Length
courierCode	String	Y	Courier Code of the shipment	10
orderIds	List<String>	Y	List of order ids associated with the shipment	20(length of each order id)

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Sample Request


```
{
    "courierCode": "ML",
    "orderIds": [
        "109678851",
        "113172764",
        "209398789",
        "116466617",
        "209398789",
        "209650705",
        "7557008",
        "7557007"
    ]
}
```

Sample Response	**For Valid Order IDs**
	
	{
	"status": {
	"statusCode": 1009,
	"statusMessage": "Shipments Verified",
	"statusType": "SUCCESS"
	},
	"courierCode": "ML",
	"orderIds": [
	"1234566789",
	"234567891"
	]
	}
	
	For Invalid Order IDs
	
	{
	"status": {
	"statusCode": 2010,
	"statusMessage": "Invalid Order IDs",
	"statusType": "ERROR"
	},
	"courierCode": "ML",
	"orderIds": [
	"1234566789",
	"234567891"
	]
	}

Status Codes	

Status Code	Status Type	Case
1009	SUCCESS	Shipments Verified
2009	ERROR	Error while validating shipments (e.g. Invalid order ids)
2010	ERROR	Invalid order ids

Get Manifest

Endpoint	https://partnerapi.myntra.com/seller/v2/logistics/getManifestFile
Http Method	POST
HttpHeader	Authorization: Bearer <Access-Token>

Description	API to be used to get manifest file against courier code and order ids.
Request Params	

Param	DataType	Required	Description	Field Length
courierCode	String	Y	Courier Code of the shipment	10
orderIds	List<String>	Y	List of order ids associated with the shipment	20(length of each order id)

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Sample Request

```
{
    "courierCode": "ML",
    "orderIds": [
        "109678851",
        "113172764",
        "209398789",
        "116466617",
        "209398789",
        "209650705",
        "7557008",
        "7557007"
    ]
}
```

Sample Response	****For Valid Order IDs****
	
	Manifest File
	
	****For Invalid Order IDs****
	
	{
	"status": {
	"statusCode": 2010,
	"statusMessage": "Invalid Order IDs",
	"statusType": "ERROR"
	},
	"courierCode": "ML",
	"orderIDs": [
	"1234566789",
	"234567891"
	]
	}

Status Codes	**Status Code** / **Status Type** / **Case** 2009 / ERROR / Missing Inputs 2010 / ERROR / Invalid order ids It will be just manifest file in case of SUCCESS.

Tracking Updates

Description	This API will be used to update the status of shipments for an order.
Endpoint	https://partnerapi.myntra.com/logistics/shipment/tracking/update
HTTP Method	POST
HTTP Header	Authorization: <authToken> Content-Type : "application/json" Accept : "application/json"

Request Body

Parameter	Data Type	Mandatory	Description
shipmentId	String	Y	Mandatory for updates from other partners. This is the order id.

shipmentType	ENUM	Y	{
			DL // Normal Forward Shipments i.e. seller to customer
			}

eventTime	DateTime	Y	Format: yyyy-MM-dd HH:mm:ss

event	Enum	Y	{
			IN_TRANSIT,
			LOST
			}

eventLocation	String	Y	maxLength: 256 characters
remarks	String	N	maxLength: 512 characters

Sample Update Request:

```
{
 "shipmentId":12323233,
 "shipmentType" : "DL",
 "event" : "OUT_FOR_DELIVERY",
 "eventTime" : "2016-11-17 10:11:01",
 "eventLocation" : "Bangalore",
 "remarks" : "Shipment is out for delivery."
}
```

Response Params

Param	Datatype	Description
statusMessage	string	Success message or appropriate error message in case of an error
statusType	string	Will denote if the request was completed successfully or not
statusType	string	Will denote if the request was completed successfully or not

HTTP Status Code

SUCCESS: 200

UNAUTHORISED: 401

ERROR: Any other code

Sample Success Response	End user needs to check statusType in the response.
	
	{
	"status": {
	"statusCode": 3,
	"statusMessage": "Success",
	"statusType": "SUCCESS",
	"totalCount": 0
	},
	"data": null
	}

Error Scenarios	End user will need to perform retries in case of error.

Outbound Calls

These are the API which Myntra will invoke on the seller systems.

Order

Create Order

Endpoint	https://<seller-domain>/v2/order/{orderId} - As provided by seller
Http Method	POST
HttpHeader	As per seller authentication rules
Description	To push an order from Myntra marketplace to the seller systems

Request Params

Param	DataType	Description	Field Length
Order	Object	Contains all the fields listed below	
id	string	The order Id, this will be the identified for every seller order captured by myntra and will be required in any order update calls there after.	20
orderLineEntries[]	List	Contains the list of order line items. Each order item structure has the highlighted properties	
id	long	The identifier corresponding to each one of the line items in the order.	20
orderId	string	The orderId of this order line	20
sku	string	The seller SKU code for which order was placed.	128
totalQuantity	int	The quantity of the item purchased	11
cancelledQuantity	int	The quantity of this line which has been cancelled	11
mrp	double	The mrp of the line item	12,2
totalSellerDiscount	double	Seller funded discount	12,2
totalMarketplaceDiscount	double	Marketplace funded discount	12,2
totalPaymentReceived	double	Payment already collected from the customer for this line	12,2
cancellationReason	string	one of CustomerInitiated, Myntra initiated or seller failed to supply	128
customizedMessage	string	Any extra information as JSON Object. Eg: For gift personalization - gift card service id can be passed as a attribute.	Free Text
shippingCharge	Double	The shipping charge applicable on this order	12,2
giftCharge	Double	Gift charges applicable on the order	12,2
amountDue	Double	Amount to be paid by the customer, post all the discounts and payments	12,2
warehouse	String	The seller warehouse code to which the order is assigned	50
receiverName	string	The name of the receiver of the order	128
address	string	The first two lines of shipping address manually entered by the customer	Free Text
city	string	The customer city	64
locality	string		255
state	string		64
country	string		64
zipCode	string		10
mobile	string	The mobile number provided by the customer	64
orderDate	Date	The order creation date	format : "yyyy-MM-dd HH:mm:ss"
cancellationReason	String	one of CustomerInitiated, Myntra initiated or seller failed to supply	128
onHold	bool	This flag indicates if the order was placed on hold on our end for some reason	1
shippingMethod	string	Either normal, SameDayDelivery or NextDayDelivery	100
expectedDeliveryTime	Date	The expected delivery date to the customer	format : "yyyy-MM-dd HH:mm:ss"
processingStartTime	Date	The time after which warehouse processing can start for this order	format : "yyyy-MM-dd HH:mm:ss"
processingCutOffTime	Date	The time before which processing has to be complete and order needs to be available for dispatch	format : "yyyy-MM-dd HH:mm:ss"
paymentMethod	String	on for online and cod for cod	10
courierCode	string	The courier code to which this order has been assigned.	10

Response Params

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Request	{
	"id": 123234455,
	"orderLineEntries": [
	{
	"id": 34444432,
	"orderId": 123234455,
	"sku": "MADU1234",
	"totalQuantity": 2,
	"cancelledQuantity": 0,
	"mrp": 400,
	"totalSellerDiscount": 100,
	"totalMarketplaceDiscount": 40,
	"totalPaymentReceived": 60,
	"cancellationReason": "",
	"customizedMessage": ""
	}
	],
	"shippingCharge": 100,
	"giftCharge": 50,
	"amountDue": 750,
	"receiverName": "customer",
	"address": "customer address",
	"city": "Bengaluru",
	"locality": "hsr layout",
	"state": "KA",
	"zipcode": "5600035",
	"country": "IN",
	"mobile": "9885533222",
	"warehouse": "MF",
	"cancellationReason": "",
	"orderDate": "2016-02-05 05:30:00",
	"onHold": "false",
	"paymentMethod": "on",
	"courierCode": "ML",
	"shippingMethod": "Normal",
	"expectedDeliveryTime": "2016-02-07 05:30:00",
	"processingStartTime;": "2016-02-06 05:30:00",
	"processingCutoffTime": "2016-02-06 16:30:0000"
	}

Sample Response	{
	"status": {
	"statusCode": 1006,
	"statusMessage": "Order created successfully",
	"statusType": "SUCCESS"
	}
	}

Status Codes	

Status Code	Status Type	Case
1006	SUCCESS	Order created successfully. (Return success in case of duplicate order as well.)
3001	ERROR	Error while creating order

Update Order

Endpoint	https://<seller-domain>/v2/order/{orderId}/<EventType>
Http Method	PUT
HttpHeader	As per seller authentication rules
Description	To push an order update from Myntra marketplace to the seller systems
Path Param	EventTypes - ItemCancellation,onhold, unhold, addressChange, shipped, delivered, trackingNumberUpdate
Request Params	Will contain the appropriate fields of the order object as per the event type. Please refer to the sample request corresponding to each EventType
Response Params	

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

Sample Request	**ItemCancellationUpdate**

Endpoint	https://<seller-domain>/v2/order/12322332/itemCancellation/

Payload	{
	"eventTime": "2016-02-06 16:30:0000",
	"orderLineEntries": [
	{
	"id": 232232323,
	"totalQuantity": 10,
	"cancelledQuantity": 1,
	"cancellationReason": "Customer Initiated / Seller SLA Breached"
	}
	]
	}

Order Place On Hold

Endpoint	https://<seller-domain>/v2/order/12322332/onhold

Payload	{
	"eventTime": "2016-02-06 16:30:0000"
	}

On Hold Resolved:

Endpoint	https://<seller-domain>/v2/order/12322332/unhold

Payload	{
	"eventTime": "2016-03-02 15:30:0009"
	}

Address Change Update:

Endpoint	https://<seller-domain>/v2/order/12322332/addressChange

Payload	{
	"eventTime": "2016-02-06 16:30:0000",
	"receiverName": "customer",
	"address": "customer address",
	"city": "Bengaluru",
	"locality": "hsr layout",
	"state": "KA",
	"zipcode": "5600035",
	"country": "IN",
	"mobile": "9885533222"
	}

Shipped:

Endpoint	https://<seller-domain>/v2/order/12322332/shipped

Payload	{
	"eventTime": "2016-02-06 16:30:0000"
	}

Delivered:

Endpoint	https://<seller-domain>/v2/order/12322332/delivered

Payload	{
	"eventTime": "2016-02-06 16:30:0000"
	}

TrackingNumber updated:

Endpoint	https://<seller-domain>/v2/order/12322332/trackingNumberUpdate

Payload	{
	"eventTime": "2016-02-06 16:30:0000"
	"trackingNumber": "342343243"
	}

Generic Update:

Endpoint	https://<seller-domain>/v2/order/123234455/update
Payload	{ "id": 123234455,

```
  "orderLineEntries": [
    {
      "id": 34444432,
      "orderId": 123234455,
      "sku": "MADU1234",
      "totalQuantity": 2,
      "cancelledQuantity": 0,
      "mrp": 400,
      "totalSellerDiscount": 100,
      "totalMarketplaceDiscount": 40,
      "totalPaymentReceived": 60,
      "cancellationReason": "",
      "customizedMessage": ""
    }
  ],
  "shippingCharge": 100,
  "giftCharge": 50,
  "amountDue": 750,
  "receiverName": "customer",
  "address": "customer address",
  "city": "Bengaluru",
  "locality": "hsr layout",
  "state": "KA",
  "zipcode": "5600035",
  "country": "IN",
  "mobile": "9885533222",
  "warehouse": "MF",
  "cancellationReason": "",
  "orderDate": "2016-02-05 05:30:00",
  "onHold": "false",
  "shippingMethod": "Normal",
  "expectedDeliveryTime": "2016-02-07 05:30:00",
  "paymentMethod": "cod",
  "processingStartTime;": "2016-02-06 05:30:00",
  "processingCutoffTime": "2016-02-06 16:30:0000"
}
```

Fields Description	Same as of in create api.

Sample Response	{
	"status": {
	"statusCode": 1008,
	"statusMessage": "Order updated successfully",
	"statusType": "SUCCESS"
	}
	}

Status Codes	

Status Code	Status Type	Case
1008	SUCCESS	Order update successfully
3002	ERROR	Error while updating order

Download Invoice

Endpoint	https://<seller-domain>/v2/myntra/downloadinvoice/{orderId}
Http Method	GET
HttpHeader	As per seller authentication rules
Description	The invoice api returns a PDF file as byte stream for the requested orderId
Path Param	orderId
Request Params	NA
Response Params	**Success :** PDF byte stream with Content-Type "application/pdf" **Error:**

Param	DataType	Description
StatusCode	int	Refer to the status code section below
StatusType	ENUM	SUCCESS, ERROR
StatusMessage	string	Will have either success or the appropriate error message in case of error.

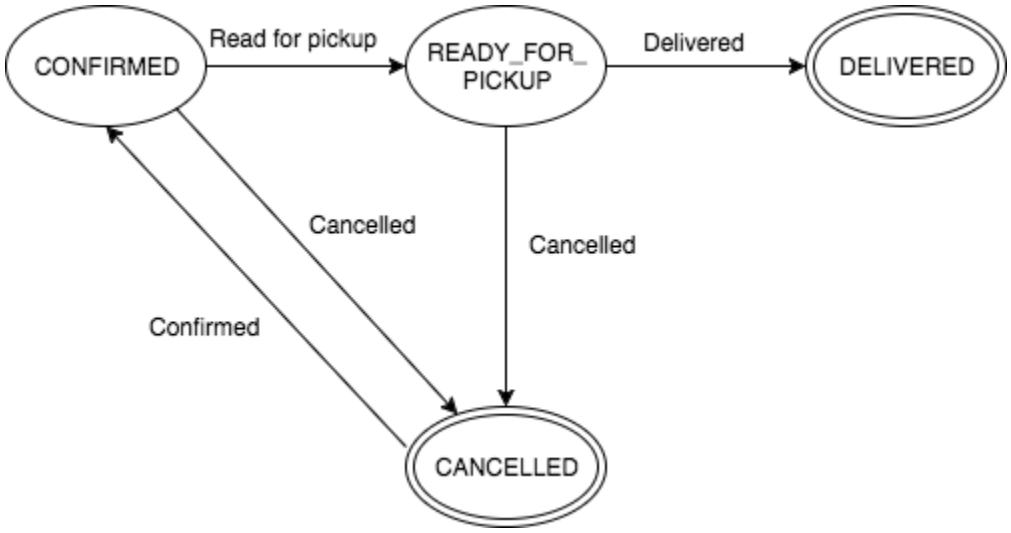
Status Code	Status Type	Case
4006	ERROR	No Invoice to download for this Order
200	SUCCESS	Invoice generated successfully

Sample Request	https://<seller-domain>/v2/myntra/downloadinvoice/70063859
Success Response	Success response will return PDF byte array with http status code with 200.

Error Response	{
	"status": {
	"statusCode": 4006,
	"statusMessage": "No Invoice to download for this Order",
	"statusType": "ERROR"
	}
	}

Return

Return state transition diagram



Create Return

Endpoint	https://<seller-domain>/v2/return/{id}
Path Params	id - Return Id
Http Method	POST
Authentication	As per seller authentication rules
Description	To create returns/RTO in seller's system. This call should be **idempotent**. Any call repeated should give same response.
Request Body	JSON

Sample Request


```
{
    "id": "123456",
    "type": "COURIER_RETURN/CUSTOMER_RETURN",
    "status": "CONFIRMED",
    "orderId": "56789",
    "createdOn": "2017-02-23 01:00:00",
    "items": [
        {
            "sku": "seller_skuA",
            "quantity": 1,
            "reason": "return reason"
        },
        {
            "sku": "seller_skuB",
            "quantity": 1,
            "reason": "return reason"
        }
    ]
}
```


Request Object Explained

parameter name	data type	description	Field Length
id	String	Unique Identifier of Return.	24
type	Enum	COURIER_RETURN, CUSTOMER_RETURN describes return type PS: COURIER_RETURN is RTO	128
status	Enum	Specifies the status of this Return. Will always be CONFIRMED in this request.	128
orderId	String	Original order id to which this return belongs	20
createdOn	Date	Specifies the time at which this return was created	
items	List<Items>	Contains the list of items in this return	
items[index].sku	String	Seller Sku code	128
items[index].quantity	Integer	Total return quantity of this sku	11
items[index].reason	String	Reason for the return of this sku	128

Response Format


```
{
    "status": {
        "statusCode": 1006,
        "statusMessage": "Return created successfully",
        "statusType": "SUCCESS"
    }
}
```

Response Fields

Param	DataType	Description
StatusCode	int	1006, 3001
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Status Code	Status Type	Case
1006	SUCCESS	Return created successfully
3001	ERROR	Error while creating return

Update Return

Endpoint	https://<seller-domain>/v2/return/{id}/update
Path Params	id - Return Id
Http Method	PUT
Authentication	As per seller authentication rules
Description	To update return in Seller's system
Request Body	JSON

Sample Request

Return Delivered

```
{
    "id": "123456",
    "type": "CUSTOMER_RETURN",
    "status": "DELIVERED"
}
```

Return Cancelled

```
{
    "id": "123456",
    "type": "CUSTOMER_RETURN",
    "status": "CANCELLED"
}
```

Return Ready For Pickup

```
{
    "id": "123456",
    "type": "CUSTOMER_RETURN",
    "status": "READY_FOR_PICKUP"
}
```

Request Object Explained

parameter name	data type	description	Field Description
id	String	This filed represents the unique Return Id. Identifier for Return Id.	24
type	Enum	COURIER_RETURN, CUSTOMER_RETURN describes return type	128
status	Enum	DELIVERED, CANCELLED, READY_FOR_PICKUP specifies the status of the return.	128

Response Format


```
{
  "status": {
    "statusCode": 1008,
    "statusMessage": "Return updated successfully",
    "statusType": "SUCCESS"
  }
}
```


Response Fields

Param	DataType	Description
StatusCode	int	1008, 3002
StatusMessage	String	Will have either success or the appropriate error message in case of error.
StatusType	Enum	SUCCESS, ERROR

Status Code	Status Type	Case
1008	SUCCESS	Return update successfully
3002	ERROR	Error while updating return

Status Code Summary:

SUCCESS:

Status Code	Case
1000	`Success`
1001	`Inventory processed successfully`
1002	`Inventory retrieved successfully`
1003	`Order updated successfully`
1004	`Items cancelled successfully`
1005	`Order retrieved successfully`
1006	`Discounts processed successfully`
1009	`Shipments verified successfully`

ERROR:

Status Code	Case
401	`Unauthorized`
2000	Error
2001	Error updating inventory
2002	Error retrieving inventory
2003	Error updating discount
2004	Error updating order
2005	Operation already performed
2006	Invalid input
2007	Invalid sku
2008	Seller item info not found
2020	Error retrieving order
2031	Order line mismatch
2032	Order line quantity mismatch
2033	Order already acknowledged
2034	Seller mismatch