



OmnichannelInventoryService Class

Route orders to inventory locations in Order Management.

Namespace

[ConnectApi](#)

OmnichannelInventoryService Methods

These methods are for `OmnichannelInventoryService`. All methods are static.

- [createReservation\(createReservationInputRepresentation\)](#)
Create an inventory reservation in Omnichannel Inventory.
- [fulfillReservation\(fulfillReservationInputRepresentation\)](#)
Fulfill one or more inventory reservations.
- [getInventoryAvailability\(inventoryAvailabilityInputRepresentation\)](#)
Retrieve inventory availability data for one or more products at one or more inventory locations or location groups.
- [getInventoryAvailabilityUploadStatus\(uploadId\)](#)
Retrieve the status of an inventory availability upload job.
- [getPublishLocationStructureStatus\(uploadId\)](#)
Retrieve the status of a publish location structure job.
- [publishLocationStructure\(\)](#)
Asynchronously publish information about your inventory locations and location groups to Omnichannel Inventory. The publish includes records whose `IsEnabled` and `ShouldSyncWithOci` fields are both `true`. This method returns an ID that you can use to retrieve the status of the publish job.
- [releaseReservation\(releaseReservationInputRepresentation\)](#)
Release one or more existing inventory reservations to free up that inventory.
- [submitInventoryAvailabilityUpload\(fileUpload\)](#)
Upload an inventory availability data file to Omnichannel Inventory.
- [transferReservation\(transferReservationInputRepresentation\)](#)
Transfer one or more inventory reservations between locations or location groups. This API doesn't change physical quantities, but reduces the reserved quantity at the source and increases it at the destination.
- [updateReservation\(updateReservationInputRepresentation\)](#)
Updates an existing reservation in Omnichannel Inventory. Add, remove, and update quantities for existing SKUs in the reservation.

createReservation(createReservationInputRepresentation)

Create an inventory reservation in Omnichannel Inventory.

API Version

51.0

Requires Chatter

No



Parameters

createReservationInputRepresentation

Type: [ConnectApi.OCICreateReservationInputRepresentation](#)

Data to reserve inventory at one or more Omnichannel Inventory locations or location groups

Return Value

Type: [ConnectApi.OCICreateReservationOutputRepresentation](#)

fulfillReservation(fulfillReservationInputRepresentation)

Fulfill one or more inventory reservations.

API Version

51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIFulfillReservationOutputRepresentation  
fulfillReservation(ConnectApi.OCIFulfillReservationInputRepresentation  
fulfillReservationInputRepresentation)
```

Parameters

fulfillReservationInputRepresentation

Type: [ConnectApi.OCIFulfillReservationInputRepresentation](#)

Wraps a list of inventory reservations to fulfill.

Return Value

Type: [ConnectApi.OCIFulfillReservationOutputRepresentation](#)

getInventoryAvailability(inventoryAvailabilityInputRepresentati

Retrieve inventory availability data for one or more products at one or more inventory locations location groups.

API Version

51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation  
getInventoryAvailability(ConnectApi.OCIGetInventoryAvailabilityInputRepresentation  
inventoryAvailabilityInputRepresentation)
```

Parameters

inventoryAvailabilityInputRepresentation



Type: [ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation](#)

getInventoryAvailabilityUploadStatus(uploadId)

Retrieve the status of an inventory availability upload job.

API Version

51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation  
getInventoryAvailabilityUploadStatus(String uploadId)
```

Parameters

uploadId

Type: [String](#)

The upload ID of the upload job.

Return Value

Type: [ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation](#)

getPublishLocationStructureStatus(uploadId)

Retrieve the status of a publish location structure job.

API Version

51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIPublishLocationStructureStatusOutputRepresentation  
getPublishLocationStructureStatus(String uploadId)
```

Parameters

uploadId

Type: [String](#)

The upload ID of the publish job.

Return Value

Type: [ConnectApi.OCIPublishLocationStructureStatusOutputRepresentation](#)

publishLocationStructure()

Asynchronously publish information about your inventory locations and location groups to Omnichannel Inventory. The publish includes records whose `IsEnabled` and `ShouldSyncWithOc`



51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIPublishLocationStructureOutputRepresentation  
publishLocationStructure()
```

Return ValueType: [ConnectApi.OCIPublishLocationStructureOutputRepresentation](#)**releaseReservation(releaseReservationInputRepresentation)**

Release one or more existing inventory reservations to free up that inventory.

API Version

51.0

Requires Chatter

No

Signature

```
public static ConnectApi.OCIReleaseReservationOutputRepresentation  
releaseReservation(ConnectApi.OCIReleaseReservationInputRepresentation  
releaseReservationInputRepresentation)
```

Parameters***releaseReservationInputRepresentation***Type: [ConnectApi.OCIReleaseReservationInputRepresentation](#)

Details of one or more inventory reservations to release.

Return ValueType: [ConnectApi.OCIReleaseReservationOutputRepresentation](#)**submitInventoryAvailabilityUpload(fileUpload)**

Upload an inventory availability data file to OmniChannel Inventory.

API Version

51.0 (NDJSON), 63.0 (CSV)

Requires Chatter

No

Signature

```
public static ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation  
submitInventoryAvailabilityUpload(ConnectApi.BinaryInput fileUpload)
```

Parameters***fileUpload***



Type: [ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation](#)

Usage

To create an inventory data file, format the data as a series of NDJSON or CSV entries that represent locations and individual inventory records.

Inventory Import Data Considerations:

- Separate the top-level entries with line feeds, not commas. Each entry must be on a single line.
- When the system reads a location entry, it assigns the subsequent inventory entries to that location until it reads another location entry.
- Legacy NDJSON requires that you specify a header record for each location entry (`{ "location": "wickenburg", "mode": "UPDATE" }`). The header isn't required for a high-performance NDJSON layout or CSV file.
- Each inventory record entry requires a unique recordId. Best practice is to use a UUID. The recordId protects against importing duplicate data. The recordId is provided in NDJSON ; automatically generated for CSV.
- Each inventory record entry requires an effectiveDate.
- If provided, each futures entry requires a nonzero quantity and a future expectedDate.

Note

The file must be in NDJSON or CSV format. For larger collections, use the Commerce API or split the data into multiple files. The Commerce API accepts GZIP, NDJSON, or CSV files up to 100 MB.

This example illustrates the data format:

Note

For readability, this example shows the first few entries on multiple lines. In the import file, each location and inventory record entry must be on a single line.

```
{
  "location": "Warehouse-A", // location identifier
  "mode": "UPDATE" // must be UPDATE (other operations might be available in future releases)
}
{
  "recordId": "0a87539d-f3dd-47bc-91c7-9c752e39dbe0", // unique identifier for the inventory record
  "onHand": 10,
  "sku": "12389156",
  "effectiveDate": "2020-12-08T14:05:22.790896-07:00",
  "futures": [ // list of future restocks
    {
      "quantity": 1,
      "expectedDate": "2021-04-18T14:05:22.781-07:00"
    },
    {
      "quantity": 5,
      "expectedDate": "2021-05-18T14:05:22.781-07:00"
    }
  ],
  "safetyStockCount": 0
}
{
  "recordId": "0a87539d-f3dd-47bc-91c7-9c752e312345",
```



```

        "expectedDate": "2021-04-18T14:05:22.781-07:00"
      }
    ],
    "safetyStockCount": 0
  }
  { "location": "Warehouse-B", "mode": "UPDATE" }
  { "recordId": "0a87539d-f3dd-47bc-91c7-9c75abc123de", "onHand": 10, "sku": "12389156", "effectDate": "2021-04-18T14:05:22.781-07:00" }
  { "recordId": "0a87539d-f3dd-47bc-91c7-9c75abc98765", "onHand": 10, "sku": "93867201", "effectDate": "2021-04-18T14:05:22.781-07:00" }
}

```

transferReservation(transferReservationInputRepresentation)

Transfer one or more inventory reservations between locations or location groups. This API does not change physical quantities, but reduces the reserved quantity at the source and increases it at the destination.

API Version

51.0

Requires Chatter

No

Signature

```

public static ConnectApi.OCITransferReservationOutputRepresentation
transferReservation(ConnectApi.OCITransferReservationInputRepresentation
transferReservationInputRepresentation)

```

Parameters

transferReservationInputRepresentation

Type: [ConnectApi.OCITransferReservationInputRepresentation](#)

Wraps a list of inventory reservation transfers and specifies whether a single failure cancels the entire list.

Return Value

Type: [ConnectApi.OCITransferReservationOutputRepresentation](#)

updateReservation(updateReservationInputRepresentation)

Updates an existing reservation in Omnichannel Inventory. Add, remove, and update quantities of existing SKUs in the reservation.

API Version

61.0

Requires Chatter

No

Signature

```

public static ConnectApi.OCIUpdateReservationOutputRepresentation
updateReservation(ConnectApi.OCIUpdateReservationInputRepresentation
updateReservationInputRepresentation)

```

Parameters



Return Value

Type: [ConnectApi.OCIUpdateReservationOutputRepresentation](#)

DID THIS ARTICLE SOLVE YOUR ISSUE?

Let us know so we can improve!

[Share your feedback](#)



DEVELOPER CENTERS

- [Heroku](#)
- [MuleSoft](#)
- [Tableau](#)
- [Commerce Cloud](#)
- [Lightning Design System](#)
- [Einstein](#)
- [Quip](#)

POPULAR RESOURCES

- [Documentation](#)
- [Component Library](#)
- [APIs](#)
- [Trailhead](#)
- [Sample Apps](#)
- [Podcasts](#)
- [AppExchange](#)

COMMUNITY

- [Trailblazer Community](#)
- [Events and Calendar](#)
- [Partner Community](#)
- [Blog](#)
- [Salesforce Admin](#)
- [Salesforce Architects](#)

© Copyright 2025 Salesforce, Inc. [All rights reserved.](#) Various trademarks held by their respective owners. Salesforce, Inc. Salesforce Tower, 415 Mission Street, 3rd Floor, San Francisco, CA 94105, United States

[Privacy Information](#) [Terms of Service](#) [Legal](#) [Use of Cookies](#) [Trust](#) [Cookie Preferences](#)

[Your Privacy Choices](#) [Responsible Disclosure](#) [Contact](#)