# Mews Connector API Integration – Key Endpoints for Market Pulse

## 1. Hotel Configuration Details

**Endpoint:** configuration/get – This endpoint returns the property (enterprise) configuration details[[1]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Request). It includes the hotel’s name, location, default currency, and address information. The official documentation for this endpoint is **Get Configuration** in the Mews Connector API docs[[1]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Request).

* **Documentation URL:** *Mews Connector API – Configuration/Get* (Connector API Documentation)[[1]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Request)
* **Request Format:** POST [PlatformAddress]/api/connector/v1/configuration/get with the required authentication tokens (ClientToken, AccessToken) in the JSON body[[2]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%60). (For single-property tokens, the associated enterprise is returned by default. Portfolio tokens can specify an EnterpriseId to get a particular property’s config[[3]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Returns%20the%20enterprise%20configuration,service%20linked%20to%20the%20token).)

**Sample Response:** The response contains an Enterprise object with nested details. For example:

{  
 "Enterprise": {  
 "Id": "851df8c8-90f2-4c4a-8e01-a4fc46b25178",  
 "Name": "Connector API Hotel",  
 "Currencies": [  
 { "Currency": "GBP", "IsDefault": true },  
 { "Currency": "USD", "IsDefault": false }  
 ],  
 "TimeZoneIdentifier": "Europe/Budapest",  
 "Address": {  
 "Line1": "I.P. Pavlova 5",  
 "City": "Prague",  
 "PostalCode": "1200",  
 "CountryCode": "CZ",  
 "Latitude": 50.075181,  
 "Longitude": 14.429645  
 }  
 // ...additional fields omitted for brevity  
 }  
}

In this JSON:  
- **Property Name** is given by Enterprise.Name (e.g. "Connector API Hotel" in the sample)[[4]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=,null).  
- **City** is part of the address, at Enterprise.Address.City (e.g. "Prague")[[5]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,14.429645).  
- **Currency Code** is found in the Enterprise.Currencies list – the default currency is marked by "IsDefault": true (e.g. "GBP" as the default currency in this example)[[6]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=,false).  
- **Address & Geolocation:** The Enterprise.Address object provides the street address (Line1, etc.), as well as Latitude and Longitude coordinates[[7]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,50.075181). For instance, the sample shows the address *I.P. Pavlova 5, Prague CZ* and coordinates (50.075181, 14.429645) corresponding to the property’s location.

## 2. Daily Performance Metrics (Daily Sync)

**Objective:** Retrieve key performance data for a specific date (“stay\_date”), including **Total Room Revenue** (excluding taxes/fees), **Rooms Sold**, and **Hotel Capacity**.

**Endpoint:** The Mews API does not provide a single “daily stats” report endpoint; instead, this data can be obtained by combining the **Reservations** and **Accounting (Order Items)** endpoints:

* Use reservations/getAll (Reservations endpoint) to get all room nights for the target date (for *Rooms Sold* and occupancy figures).
* Use orderItems/getAll (Accounting endpoint) to fetch revenue items (for *Total Room Revenue* on that date).
* Hotel *Capacity* (total rooms) can be retrieved from the property configuration or by counting resources via resources/getAll if needed.

**Reservations (Rooms Sold):** The reservations/getAll operation can filter reservations by date. By specifying a one-day interval in the request, you can retrieve all reservations that overlap that date (these represent rooms occupied/sold on that date)[[8]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%22StartUtc%22%3A%20%222016,abbe01240d9c%22%20%5D%2C%20%22CustomerIds%22%3A)[[9]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=Reservation%20time%20filter). This is done using the **TimeFilter** and date range parameters in the JSON body:

* **Date Filter:** Set "StartUtc" to the start of the date and "EndUtc" to the end of the date (exclusive of the next day at 00:00 UTC). For example, to get data for **2025-08-01**, use "StartUtc": "2025-08-01T00:00:00Z" and "EndUtc": "2025-08-02T00:00:00Z".
* **TimeFilter:** Use "TimeFilter": "Colliding" to fetch any reservation that **collides with** (overlaps) the interval[[9]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=Reservation%20time%20filter). This ensures that all stays that cover that night are returned (not just arrivals or departures). (If instead you wanted only check-ins on that date, you could use "TimeFilter": "Start" for arrivals in that interval[[10]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%2A%20%60Updated%60%20,within%20the%20specified%20interval).)
* **ServiceIds:** Include the ID of the Accommodation service to filter to lodging reservations.
* **Extent:** Optionally, set Extent flags to include related data (e.g. Reservations: true, Resources: true to get room details, etc.)[[11]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=,ae8300be8ab7%22%2C%20%22Count%22%3A%2010). You could also set OrderItems: true to fetch associated charges with each reservation, though it may be more straightforward to use the Order Items endpoint separately.

**Example Request (reservations/getAll for one date):**

POST [PlatformAddress]/api/connector/v1/reservations/getAll  
{  
 "ClientToken": "...",   
 "AccessToken": "...",  
 "ServiceIds": [ "bd26d8db-86da-4f96-9efc-e5a4654a4a94" ], // Accommodation service ID  
 "StartUtc": "2025-08-01T00:00:00Z",  
 "EndUtc": "2025-08-02T00:00:00Z",  
 "TimeFilter": "Colliding",  
 "Extent": { "Reservations": true, "Resources": true }  
}

This will return all reservations that were in-house on 2025-08-01. From the response, you can count the number of reservations or occupied rooms (Rooms Sold). If the Resources extent is included, each reservation’s assigned room (AssignedResourceId) can be used to ensure each physical room is counted once. The total number of available rooms (Capacity) can be obtained by calling resources/getAll (which lists all room resources in the property) and counting them[[7]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,50.075181) or by reading the Enterprise.Configuration (which may include number of resources or can be derived by resource list).

**Accounting (Total Room Revenue):** To get revenue, use the orderItems/getAll endpoint. This returns all charge items (financial transactions) posted in Mews. You can filter these by date and type to isolate room revenue for the target date:

* **Date Filter:** Use the ConsumedUtc filter to specify the date range for consumption (the date the service was rendered or the revenue is recognized). For a given stay date, set ConsumedUtc.StartUtc and ConsumedUtc.EndUtc to that date’s boundaries[[12]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22ConsumedUtc%22%3A%20%7B%20%22StartUtc%22%3A%20%222023,31T00%3A00%3A00Z%22). For example, for 2025-08-01 use "ConsumedUtc": { "StartUtc": "2025-08-01T00:00:00Z", "EndUtc": "2025-08-02T00:00:00Z" }.
* **Type Filter:** Optionally filter by AccountingCategoryId or Order item Type to include only room revenue. Room charges typically fall under a specific accounting category (e.g. Accommodation). You may exclude items like taxes or fees. (In the response, each OrderItem includes a breakdown of Net and Tax values[[13]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22AccountingCategoryId%22%3A%20%22c0610937,Z%22%2C%20%22Value%22%3A%200%20%7D)[[14]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=,Z%22%2C%20%22Value%22%3A%200%20%7D). Summing the NetValue of all room accommodation items for the date will give **Total Room Revenue (net of taxes)**.)

**Example Request (orderItems/getAll for one date’s revenue):**

POST [PlatformAddress]/api/connector/v1/orderItems/getAll  
{  
 "ClientToken": "...",  
 "AccessToken": "...",  
 "ServiceIds": [ "294c7859-63ba-46ad-a8bf-34fad2019383" ], // Accommodation service ID  
 "ConsumedUtc": {  
 "StartUtc": "2025-08-01T00:00:00Z",  
 "EndUtc": "2025-08-02T00:00:00Z"  
 },  
 "AccountingStates": [ "Open", "Closed" ]  
}

This will return all billing items consumed on 2025-08-01. Each item has an Amount with NetValue and TaxValues. For example, a room night item might appear with "Amount":{"NetValue": 100, "GrossValue": 110, "TaxValues":[...]}, and a separate city tax item might appear with Type: "CityTax"[[15]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=,)[[16]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22RevenueType%22%3A%20%22Additional%22%2C%20%22CreatorProfileId%22%3A%20%223cd637ef,30T22%3A00%3A00Z). By filtering or post-processing, you sum up **NetValue** of room/night items (excluding any tax or ancillary items) to get the Total Room Revenue (excl. taxes) for that date.

**Documentation References:**  
- *Reservations – Get All Reservations:* This is a **Reservations** endpoint (operational data) used for occupancy-related data[[8]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%22StartUtc%22%3A%20%222016,abbe01240d9c%22%20%5D%2C%20%22CustomerIds%22%3A)[[9]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=Reservation%20time%20filter).  
- *Order Items – Get All Order Items:* This is an **Accounting** endpoint used for financial data like revenue[[17]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22ConsumedUtc%22%3A%20%7B%20%22StartUtc%22%3A%20%222023,31T00%3A00%3A00Z%22%20%7D%2C%20%22ClosedUtc%22%3A)[[18]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22AccountingCategoryId%22%3A%20%22c0610937,Z%22%2C%20%22Value%22%3A%200).

*(Note: Mews does not provide a single “daily performance” API call; an integration combines the above data. Also, ensure to use the property’s local time as needed – Mews timestamps are in UTC by default.)*

## 3. Historical Performance Metrics (Initial Sync)

When first integrating, you may need to pull **historical data** (e.g. a year’s worth of daily metrics). The same endpoints from (2) can be used for larger date ranges, with some considerations for range limits and pagination:

* **Date Range Support:** The reservations/getAll endpoint **does** support specifying a date interval (StartUtc/EndUtc as shown above) to retrieve reservations in that range[[8]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%22StartUtc%22%3A%20%222016,abbe01240d9c%22%20%5D%2C%20%22CustomerIds%22%3A). However, for very large ranges, it’s recommended to break the query into smaller chunks (to avoid timeouts or overly large responses)[[19]](https://mews-systems.gitbook.io/connector-api/guidelines/requests#:~:text=In%20rare%20circumstances%2C%20you%20may,a%20mitigation%20solution%20in%20place)[[20]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=The%20RMS%20should%20fetch%20the,would%20give%20the%20RMS%20all). Mews’ documentation suggests fetching reservations in batches (for example, week-by-week or month-by-month) for a year of data rather than one massive request[[20]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=The%20RMS%20should%20fetch%20the,would%20give%20the%20RMS%20all). The orderItems/getAll endpoint supports up to 3 months in a single query for ConsumedUtc range (as indicated by “max length 3 months” in the docs)[[21]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=), so historical revenue should also be pulled in chunks (e.g. quarter by quarter).
* **Pagination:** Both endpoints use **cursor-based pagination** for large result sets[[22]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=Returns%20all%20order%20items,and%20supports%20%2011)[[23]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Pagination%20is%20a%20feature%20introduced,This%20takes%20the%20following%20form). This means you can limit the number of records per call and use a cursor token to fetch subsequent pages. In the request JSON, you include a Limitation object with a Count (max items to retrieve) and an optional Cursor[[11]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=,ae8300be8ab7%22%2C%20%22Count%22%3A%2010)[[24]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%22E0D439EE522F44368DC78E1BFB03710C,ae8300be8ab7%22%2C%20%22Count%22%3A%2010%20%7D). If Cursor is empty or not provided on the first call, the API returns the first page (most recent items by default). The response will include a Cursor field (token) pointing to the next page of older data[[25]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Count%20of%20items%20to%20be,returned%2C%20minimum%201%2C%20maximum%201000)[[26]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%7B%20,ae51010a5aa4%22). You then repeat the call, supplying the returned cursor to get the next batch, and so on, until all data in the range is retrieved.
* **Example (Pagination):** You might start with a request: { StartUtc: "2024-01-01T00:00:00Z", EndUtc: "2024-03-01T00:00:00Z", Limitations: { Count: 1000 } } to get January–February 2024 data (up to 1000 records). The response will return a Cursor (e.g. "Cursor": "7f9325f6-ef44-4911-89a8-ae51010a5aa4" in the JSON)[[26]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%7B%20,ae51010a5aa4%22) if there are more than 1000 records. For the next call, include that cursor: "Limitation": { "Cursor": "7f9325f6-ef44-4911-89a8-ae51010a5aa4", "Count": 1000 } to get the next page[[25]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Count%20of%20items%20to%20be,returned%2C%20minimum%201%2C%20maximum%201000). Continue until no further cursor is returned or until you reach your date range end. This approach is needed for **historical reservations** and **order items** if the dataset is large.
* **Alternative Bulk Export:** Mews also offers a **Bulk Data Export** feature as an alternative to pulling many pages of data. Instead of multiple incremental calls, you can use the *Add Export* endpoint to request an export of all reservations (or other data) for a given period, and then retrieve the result file[[27]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=Bulk%20data%20export). For initial sync of very large historical datasets, this can be useful. (See **Add Export** / **Get all exports** in the documentation[[28]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=As%20an%20alternative%20to%20making,a%20bulk%20%2021%20feature)[[29]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=How%20to%20check%20on%20data,export%20progress).)

**Documentation for Historical Data & Pagination:** Refer to the **Pagination** section of the Mews Connector API docs for details on cursor-based pagination[[25]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Count%20of%20items%20to%20be,returned%2C%20minimum%201%2C%20maximum%201000)[[26]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%7B%20,ae51010a5aa4%22). The **Revenue Management** use-case guide in Mews docs also discusses strategies for initial data pulls and syncing (e.g. batching weekly)[[20]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=The%20RMS%20should%20fetch%20the,would%20give%20the%20RMS%20all). These resources explain how to safely fetch large date ranges and handle pagination tokens.

## 4. API Environments (Platform Addresses)

Mews provides two main environments for its Connector API: **Demo** (test environment) and **Production** (live environment). The base URL (PlatformAddress) differs for each:

| Environment | PlatformAddress Base URL |
| --- | --- |
| **Production** | https://api.mews.com[[30]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Addresses) |
| **Demo (Sandbox)** | https://api.mews-demo.com[[31]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=applications%3A) |

Developers should direct API calls to the appropriate base address depending on context. The Demo environment is used for development/testing and uses test data (it should **not** contain real customer data)[[32]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Mews%20supports%20two%20main%20environments%3A,is%20for%20live%20customer%20sites)[[33]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Security%20Policy). The Production environment is for live data and requires production credentials/tokens[[34]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Security%20Policy)[[35]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=API%20tokens). Aside from the URL, the API endpoints paths are the same (e.g. /api/connector/v1/...), and the Demo environment has both Gross and Net tax mode variants for testing different tax configurations[[36]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Pricing%20environments).

**Documentation Reference:** See the **Environments** section of the Mews API documentation[[37]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Platform%20addresses)[[30]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Addresses), which lists the PlatformAddress for Demo and Production, as well as other environment details. This section confirms: *“PlatformAddress - https://api.mews-demo.com”* for Demo and *“PlatformAddress - https://api.mews.com”* for Production[[31]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=applications%3A)[[30]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Addresses).

**Sources:**

* Mews Connector API Documentation – *Configuration (Get configuration)*[[1]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Request)[[7]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,50.075181)
* Mews Connector API Documentation – *Reservations (Get all reservations)*[[8]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%22StartUtc%22%3A%20%222016,abbe01240d9c%22%20%5D%2C%20%22CustomerIds%22%3A)[[9]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=Reservation%20time%20filter)
* Mews Connector API Documentation – *Order Items (Get all order items)*[[12]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22ConsumedUtc%22%3A%20%7B%20%22StartUtc%22%3A%20%222023,31T00%3A00%3A00Z%22)[[14]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=,Z%22%2C%20%22Value%22%3A%200%20%7D)
* Mews Connector API Documentation – *Pagination Guidelines*[[25]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Count%20of%20items%20to%20be,returned%2C%20minimum%201%2C%20maximum%201000)[[26]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%7B%20,ae51010a5aa4%22)
* Mews Connector API Documentation – *Environments (Demo vs Production)*[[31]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=applications%3A)[[30]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Addresses)
* Mews “Revenue Management” Guide – *Initial data pull and batching recommendations*[[20]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=The%20RMS%20should%20fetch%20the,would%20give%20the%20RMS%20all).

[[1]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Request) [[2]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%60) [[3]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=Returns%20the%20enterprise%20configuration,service%20linked%20to%20the%20token) [[4]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=,null) [[5]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,14.429645) [[6]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=,false) [[7]](https://mews-systems.gitbook.io/connector-api/operations/configuration#:~:text=%22Address%22%3A%20%7B%20%22Id%22%3A%20%228c2c4371,50.075181) Configuration | Connector API

<https://mews-systems.gitbook.io/connector-api/operations/configuration>

[[8]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%22StartUtc%22%3A%20%222016,abbe01240d9c%22%20%5D%2C%20%22CustomerIds%22%3A) [[9]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=Reservation%20time%20filter) [[10]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=%2A%20%60Updated%60%20,within%20the%20specified%20interval) [[11]](https://mews-systems.gitbook.io/connector-api/operations/reservations#:~:text=,ae8300be8ab7%22%2C%20%22Count%22%3A%2010) Reservations | Connector API

<https://mews-systems.gitbook.io/connector-api/operations/reservations>

[[12]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22ConsumedUtc%22%3A%20%7B%20%22StartUtc%22%3A%20%222023,31T00%3A00%3A00Z%22) [[13]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22AccountingCategoryId%22%3A%20%22c0610937,Z%22%2C%20%22Value%22%3A%200%20%7D) [[14]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=,Z%22%2C%20%22Value%22%3A%200%20%7D) [[15]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=,) [[16]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22RevenueType%22%3A%20%22Additional%22%2C%20%22CreatorProfileId%22%3A%20%223cd637ef,30T22%3A00%3A00Z) [[17]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22ConsumedUtc%22%3A%20%7B%20%22StartUtc%22%3A%20%222023,31T00%3A00%3A00Z%22%20%7D%2C%20%22ClosedUtc%22%3A) [[18]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=%22AccountingCategoryId%22%3A%20%22c0610937,Z%22%2C%20%22Value%22%3A%200) [[21]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=) [[22]](https://mews-systems.gitbook.io/connector-api/operations/orderitems#:~:text=Returns%20all%20order%20items,and%20supports%20%2011) Order items | Connector API

<https://mews-systems.gitbook.io/connector-api/operations/orderitems>

[[19]](https://mews-systems.gitbook.io/connector-api/guidelines/requests#:~:text=In%20rare%20circumstances%2C%20you%20may,a%20mitigation%20solution%20in%20place) Requests | Connector API

<https://mews-systems.gitbook.io/connector-api/guidelines/requests>

[[20]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=The%20RMS%20should%20fetch%20the,would%20give%20the%20RMS%20all) [[27]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=Bulk%20data%20export) [[28]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=As%20an%20alternative%20to%20making,a%20bulk%20%2021%20feature) [[29]](https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management#:~:text=How%20to%20check%20on%20data,export%20progress) Revenue management | Connector API

<https://mews-systems.gitbook.io/connector-api/use-cases/revenue-management>

[[23]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Pagination%20is%20a%20feature%20introduced,This%20takes%20the%20following%20form) [[24]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%22E0D439EE522F44368DC78E1BFB03710C,ae8300be8ab7%22%2C%20%22Count%22%3A%2010%20%7D) [[25]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=Count%20of%20items%20to%20be,returned%2C%20minimum%201%2C%20maximum%201000) [[26]](https://mews-systems.gitbook.io/connector-api/guidelines/pagination#:~:text=%7B%20,ae51010a5aa4%22) Pagination | Connector API

<https://mews-systems.gitbook.io/connector-api/guidelines/pagination>

[[30]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Addresses) [[31]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=applications%3A) [[32]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Mews%20supports%20two%20main%20environments%3A,is%20for%20live%20customer%20sites) [[33]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Security%20Policy) [[34]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Security%20Policy) [[35]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=API%20tokens) [[36]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Pricing%20environments) [[37]](https://mews-systems.gitbook.io/connector-api/guidelines/environments#:~:text=Platform%20addresses) Environments | Connector API

<https://mews-systems.gitbook.io/connector-api/guidelines/environments>