## Introduction

This guide will cover a variety of aspects pertaining to managing inventory for products and variants. Before we begin, it is important to understand variants.

<div class="note"><strong>04/01/2017:</strong> Inventory-related data points on `Product` are being moved into a nested `Inventory` object. Learn more about the Inventory Revamp [here](https://github.com/ordercloud-api/release-notes/blob/master/2017/January/breaking-changes.md#inventory-revamp).</div>

## Prerequisites

* [Generating Variants](https://documentation.ordercloud.io/use-case-guides/product-catalog-management/generating-variants)

## 1. Enable Inventory

The first step to managing inventory at either a product or variant level is to enable this functionality. To do this, set the `InventoryEnabled` (Product level) and/or `VariantLevelInventory` (Variant level) to `true` when creating/updating your Product.

<div class="api-reference">API Reference: [Create New Product](<https://documentation.ordercloud.io/api-reference#Products_Create>)</div>

```

POST https://api.ordercloud.io/v1/products HTTP/1.1  
Authentication: Bearer put\_access\_token\_here  
Content-Type: application/json

{  
 "ID": "ProductID123",  
 "Name": "Product X",  
 "Description": "This is product X",  
 "QuantityMultiplier": 1,  
 "Active": true,  
 "InventoryEnabled": true,  
 "VariantLevelInventory": true  
}

```

## 2. Configure Inventory Settings

Once inventory is enabled for a product, you can configure some various inventory settings. There are a few other inventory related fields that can be configured when creating/updating a product: `InventoryVisible`, `AllowOrderExceedInventory`, and `InventoryNotificationPoint`.

`InventoryVisible` is for display purposes only. You can use this to determine whether or not to show a product’s current inventory levels within a buyer application.

`AllowOrderExceedInventory` allows you to configure whether or not orders can exceed available inventory for a particular product. If this is the case, it is common to display some type of messaging indicating the product is on backorder.

`InventoryNotificationPoint` is an integer value that will trigger an inventory notification message if a product’s available inventory drops below the configured threshold.

## 3. Update Inventory

Lastly, inventory can be managed at either a product or variant level using various OrderCloud endpoints. `List` and `Get` endpoints exist for both products and variants in order to determine whether inventory values are synced between OrderCloud and your own system.

Inventory counts consist of two properties: Available and Reserved. Both of these are derived values based on live data. Available is the total inventory quantity of a Product or Variant that is not currently in an open order. Reserved is the quantity of a Product or Variant that are currently included in any open orders containing that item. Once an order is submitted, the quantity ordered will be removed from Reserved and decremented from Available.

If you need to update a Product or Variant’s inventory, the total inventory quantity for that Product or Variant should be sent using the endpoints referenced above. Below is an example of how updating a Product’s inventory will affect the derived values:

<div class="api-reference">API Reference: [Update Inventory](https://documentation.ordercloud.io/api-reference#Products\_UpdateInventory)</div>

<table class="table table-bordered table-responsive">

<tr>

<th>Available</th>

<th>Reserved</th>

<th>Inventory</th>

</tr>

<tr>

<td>500</td>

<td>200</td>

<td>700</td>

</tr>

<tr>

<td>800</td>

<td>200</td>

<td>1000</td>

</tr>

</table>

## Conclusion

You should now be familiar with some of the different aspects pertaining to managing inventory for products and variants.