

## **Overview**

## **Implementation**

### **1. Setup Odoo Shopify Connector**

- 1.1 Create an API Key, Password & a Secret Key
- 1.2 Set up Shopify Instance in Odoo

### **2. Odoo Shopify Instance Configurations**

- 2.1 Currency Information
- 2.2 Product Information
- 2.3 Order Information
- 2.4 Tax Information

### **3. Webhook Configurations**

- 3.1 How to set up a Webhook

### **4. Odoo Shopify Operations**

- 4.1 Import Operations
  - a. Import Product
    - i) Using 'Import All'
    - ii) Import Using 'Specific ID'
      - A.1 Create a Product in Shopify
      - A.2 Update a Product in Shopify
  - b. Import Product Stock
  - c. Import Order
  - d. Import Customer

#### **4.2 Export Operations**

- a. Export Product
  - i) Create a Product in Odoo
    - b. Export Stock
    - c. Export Customer
    - d. Export Order

#### **4.2.1 Export Record in Bulk**

#### **4.3 Auto Configure Schedulers**

#### **4.4 Product/Order Associated Operations**

- 4.4.1 Collections

- 4.4.2 Discounts
- 4.4.3 Locations
- 4.4.4 Draft Orders

## **5. Dashboard**

## **6. Meta Mapping**

## **7. Sale Workflow**

## **8. Auto Product Sync**

## **9. Report**

### **9.1 Sale Reporting**

- a. Bar Chart
- b. Pie Chart
- c. Line Chart
- d. Pivot View

### **9.2 Instance Wise Report**

### **9.3 Email Report Configuration**

## **10. Stock Refund**

- a. Import Order
- b. Update Stock
- c. Validate Inventory
- d. Validate Delivery
- e. Generate Invoice
- f. Creating Payment
- g. Refund of Order and Product Stock

## **11. Cancel Order**

## Overview

### For Quick, Effective, and Accurate Ecommerce Operations!

Shopify is a cloud-based eCommerce platform helping small to medium size merchants globally. It helps eCommerce business owners to connect with the globe and sell their products online. Ecommerce sellers that use Odoo to manage their Shopify backend can now automate their workflow with Ksolves Odoo Shopify connector. It lets you integrate your Shopify store to Odoo to perform various import and export operations.

Import Orders, Products, Customers, and more from Shopify to Odoo, or vice versa. Use our Odoo Shopify Connector to automate Shopify Odoo Integration operations to save time and effort.

**Note:** It is necessary to install the ‘Odoo Base Connector’ in order to utilize the functionalities of the Shopify Connector.

## Implementation

Download the Odoo Shopify Connector app from the Odoo Store or Ksolves Store. After successful installation, add it to the Odoo Add-ons path or any preferred location for ease of use. Refresh the Odoo instance and go to the Apps section of the dashboard. Select ‘Update Apps List’ from the menu. Odoo Shopify Connector will appear in Apps after successful loading.

Click on the ‘Install’ button from the drop-down menu. Success! Now you are all set to utilize the Odoo Shopify Connector.

**Important:** After the Odoo Shopify connector is installed, the “Shopify connector” menu is required to be made visible, by giving the Odoo user, full access to Shopify.

1. Go to settings → Users & company (in the top menu) → (click on) Users.
2. Choose your user (e.g. admin) and open the respective record and click on Edit
3. From **Shopify, access** field choose the **full access** option and click Save
4. Refresh the browser

5. Now the Shopify connector menu will be visible.

## 1. Setup Odoo Shopify Connector

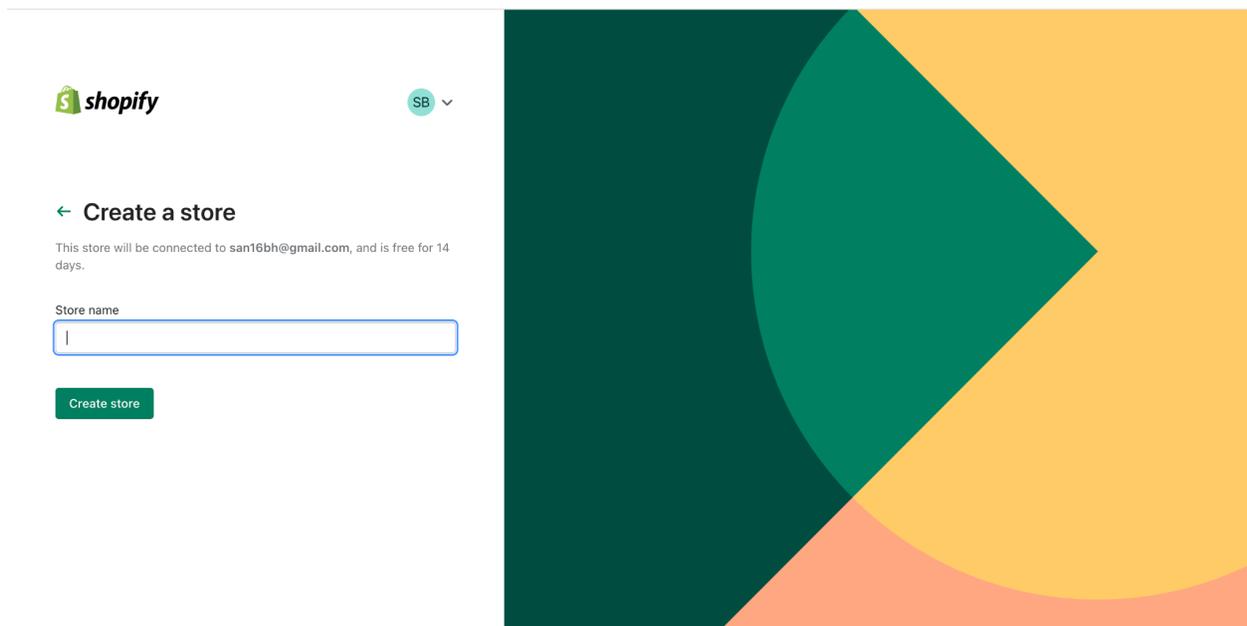
**Remember:** Anyone who is using either Shopify or Odoo for the first time, can use this connector to automate the import/export operations.

It is important to Generate the API Key, password & Secret Key in order to connect your Shopify store with Odoo ERP.

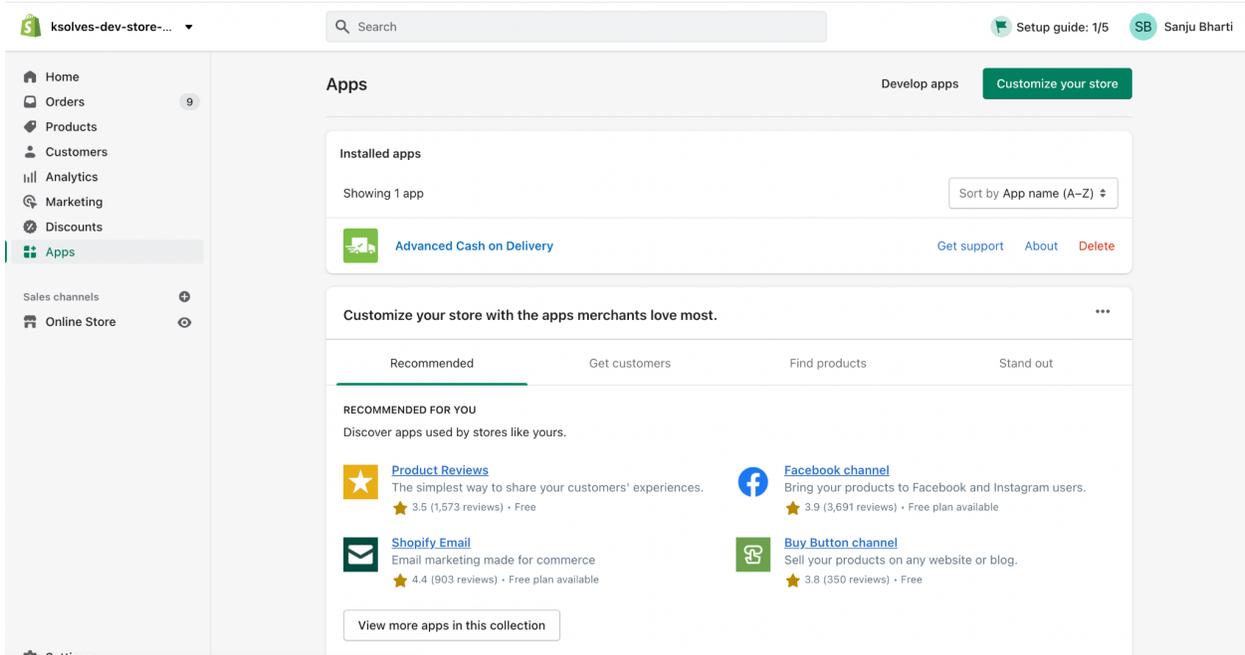
### 1.1 Extract API Key & Secret Key

#### 1.1.1 Create a Shopify Store

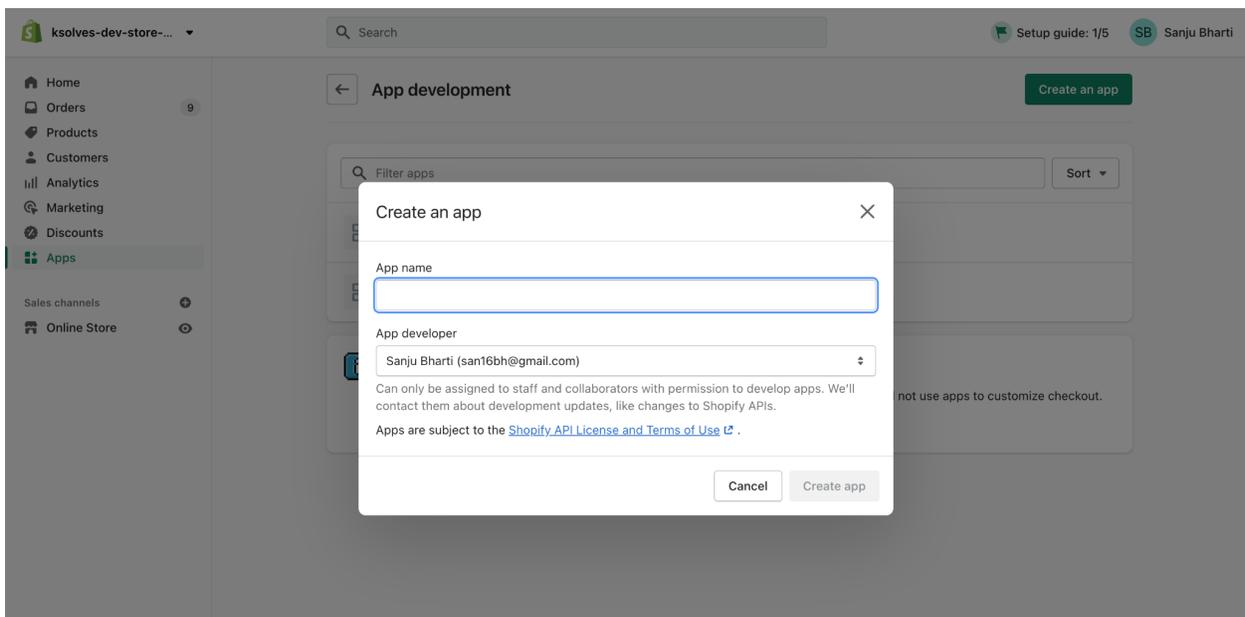
You can either log in to your already existing Shopify store or create a new one. If you are creating a new store, fill in all the details and log in to your new store.



Click on the **APPs** button and click on the **Develop Apps>Create an App**.



Give the name of the App here we have given **odoo-ksolves-store-001** and click on the **Create App** button.



Open the App and click on the Configuration button. Fill in all the details about your store and save the settings.

**Note:** You can configure these settings later as per your requirements.

Overview Configuration API credentials App settings

Configuration > Admin API integration

**Admin API access scopes**

All Selected 62

Filter access scopes

**Analytics**  
View store metrics  read\_analytics

**Assigned fulfillment orders**  
View or manage fulfillment orders  write\_assigned\_fulfillment\_orders  
 read\_assigned\_fulfillment\_orders

**Customers**  
Includes personally identifiable information  
View or manage customers, customer addresses, order history, and customer groups  write\_customers  
 read\_customers

Get access to store data with the [Admin API](#). Only select the scopes your app needs. Don't access APIs in ways that violate the [Shopify API License and Terms of Use](#).

[Explore Admin API scopes](#)

[Start using the Admin API](#)

[Run demo queries with Shopify's GraphQL App](#)

Click on the API credentials and click on the 'Install the App' button. Once the App is installed, there you will find the API Key & Secret Key, and Shared Secret Key. Copy them on a notepad so that you can use them later.

Overview Configuration API credentials App settings

API credentials

**Admin API access token** New

Use your access token to request data from the [Admin API](#).

.....\*6b18

Created 7 Mar 2022 at 13:22 UTC

You can't view this again because it provides authenticated Admin API access.

Keep your access tokens secure. Only share them with developers that you trust to safely access your data.

[Authenticate with access tokens](#)

[Start using the Admin API](#)

[Start using the Storefront API](#)

**API key and secret key**

API key

3f43f363106b3206fe3e1031997daa85

API secret key

.....

Created 7 Mar 2022 at 09:00 UTC

Use your client secret to [verify incoming webhooks](#).

## 1.2 Set up Shopify Instance in Odoo

In Odoo, separate instances are required for each Shopify store. Open Shopify Connector>Instance>Create

A new window will open that will ask you to fill in the required details in the Authentication section, including Customer Key, Customer Password, and Shared Key.

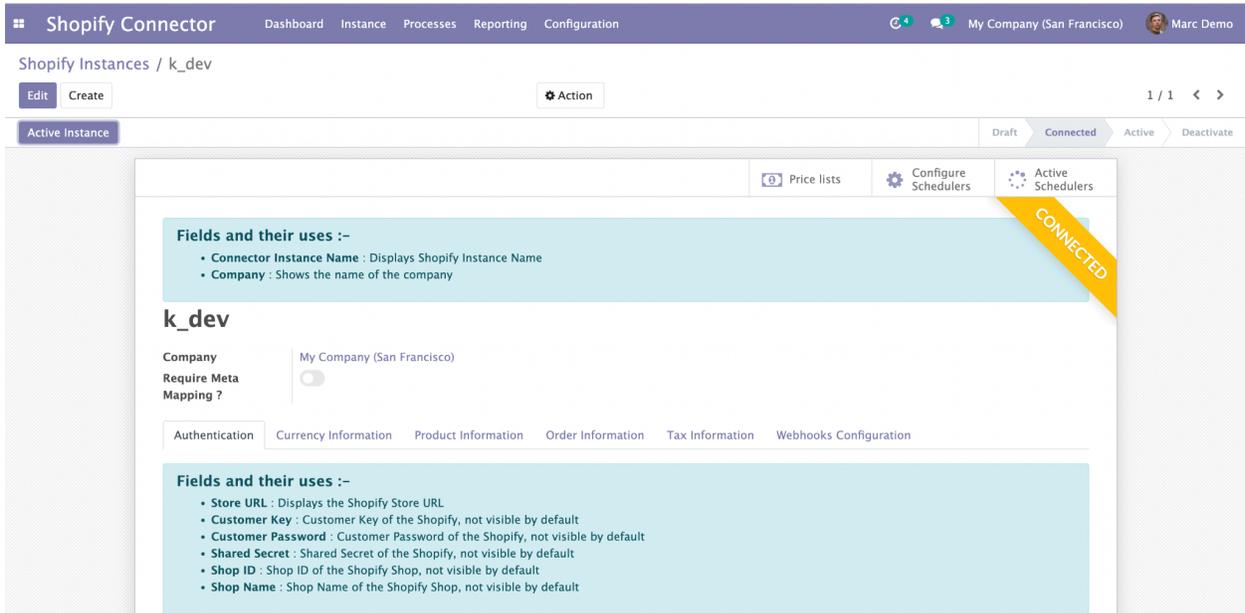
Give the instance name, fill in the necessary details, and click on save.

The screenshot shows the 'Shopify Connector' interface in Odoo. The breadcrumb trail is 'Shopify Instances / DEMO'. There are 'Save' and 'Discard' buttons at the top left. A navigation bar at the top includes 'Dashboard', 'Instance', 'Processes', 'Reporting', and 'Configuration'. The user is logged in as 'My Company (San Francisco)' with the name 'Marc Demo'. The form title is 'Shopify Instances / DEMO'. Below the title are 'Save' and 'Discard' buttons. A help box explains: 'Connector Instance Name : Displays Shopify instance name' and 'Company : Shows the name of the company'. The 'Connector Instance Name' field contains 'k\_dev'. The 'Company' dropdown is set to 'My Company (San Francisco)'. There is a 'Require Meta Mapping?' toggle switch. Below this is a tabbed interface with 'Authentication' selected, and other tabs for 'Currency Information', 'Product Information', 'Order Information', 'Tax Information', and 'Webhooks Configuration'. A 'Fields and their uses :-' section lists: 'Store URL : Displays the Shopify Store URL', 'Customer Key : Customer Key of the Shopify, not visible by default', 'Customer Password : Customer Password of the Shopify, not visible by default', 'Shared Secret : Shared Secret of the Shopify, not visible by default', 'Shop ID : Shop ID of the Shopify Shop, not visible by default', and 'Shop Name : Shop Name of the Shopify Shop, not visible by default'. The form fields are: 'Store URL' (https://ksolves-dev-store-001.myshopify.com/), 'Customer Key' (masked with dots), 'Customer Password' (masked with dots), and 'Shared Secret' (masked with dots).

### ● Connect Instance

Click on the 'connect instance' button, and you will get a pop-up window showing the connection is successful.

Once the connection is successful you will see a yellow banner at the corner of your screen showing 'Connected'.



- **Activate Instance**

It is important to activate the instance in order to make the integration between odoo and Shopify successful.

Note: To activate an instance, some mandatory fields are important to mention.

→ Go to Product Information>give Stock Field Type



→ Go to Order Information>Fill Warehouse, Sales Team, & Sales Person fields

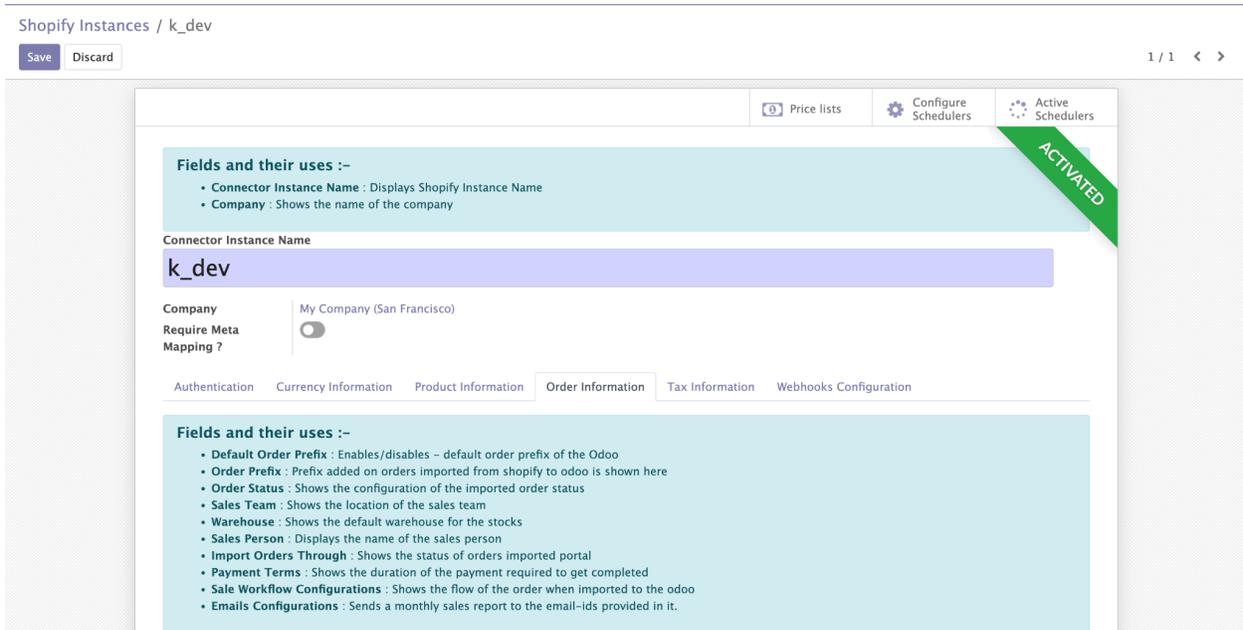
**Fields and their uses :-**

- **Default Order Prefix** : Enables/disables – default order prefix of the Odoo
- **Order Prefix** : Prefix added on orders imported from shopify to odoo is shown here
- **Order Status** : Shows the configuration of the imported order status
- **Sales Team** : Shows the location of the sales team
- **Warehouse** : Shows the default warehouse for the stocks
- **Sales Person** : Displays the name of the sales person
- **Import Orders Through** : Shows the status of orders imported portal
- **Payment Terms** : Shows the duration of the payment required to get completed
- **Sale Workflow Configurations** : Shows the flow of the order when imported to the odoo
- **Emails Configurations** : Sends a monthly sales report to the email-ids provided in it.

Default Order Prefix	<input type="checkbox"/>	Warehouse	San Francisco
Order Prefix	SC	Sales Team	Europe
Order Status		Sales Person	Joel Willis
Import Orders through	Status	Payment Term	Joel Willis
			Create and Edit...

After filling in all the required details, click on Save and ‘Activate Instance’ button.

Once the connection is successful you will see a green banner at the corner of your screen showing ‘Activated’.



## 2. Odoo Shopify Instance Configurations

Shopify configurations are mandatory to define your store in Odoo. This section covers the four configurations (information).

## 2.1 Currency Information

This tab displays the main currency and pricelists (regular and compared) that will be used for the Shopify Store on Odoo.

A user can select the currency type as per their region. Refer to the 'Field and their uses' section for detailed information.

The screenshot shows the 'Currency Information' tab for a Shopify instance named 'K\_dev'. The interface includes a 'Save' button and a 'Discard' button. A green 'ACTIVATED' banner is visible in the top right corner. The 'Fields and their uses' section lists:

- Connector Instance Name : Displays Shopify Instance Name
- Company : Shows the name of the company

The 'Connector Instance Name' field is set to 'K\_dev'. The 'Company' field is set to 'My Company (San Francisco)'. The 'Require Meta Mapping?' checkbox is unchecked. The 'Currency Information' tab is selected, with other tabs including Authentication, Product Information, Order Information, Tax Information, and Webhooks Configuration.

The second 'Fields and their uses' section lists:

- Main Currency : Displays the main currency of the shopify shop
- Regular Main Pricelist : Displays the regular pricelist of the main currency on odoo.
- Compare Main Pricelist : Displays the compare pricelist of the main currency on odoo.
- Multi-Currency : Displays the multi-currency which user can select.
- Multi-Pricelist : Displays the multi-pricelist for all the currency selected.

The configuration fields are as follows:

- Main Currency: INR
- Regular Main Pricelist: [ K\_dev ] INR Regular Pricelist (INR) (checked)
- Compare Main Pricelist: [ K\_dev ] INR Compare Pricelist (INR) (checked)
- Multi-Currency: USD (selected)
- Multi-Pricelist: [ K\_dev ] INR Compare Pricelist (...), [ K\_dev ] INR Regular Pricelist (...)

## 2.2 Product Information

This page/tab displays the parameters (to be set) using which you want to sync your products between Shopify and Odoo.

You can sync product images and prices by enabling the given checkboxes.

Select the Stock Field Type to determine that imported stock will either be an on-hand quantity or a forecasted quantity.

The screenshot displays the configuration interface for a Shopify Connector instance named 'K\_dev'. The interface includes a navigation bar at the top with options like 'Dashboard', 'Instance', 'Processes', 'Reporting', and 'Configuration'. The main content area is titled 'Shopify Instances / K\_dev' and features a 'Save' and 'Discard' button. A green 'ACTIVATED' banner is present in the top right. The configuration is organized into tabs: 'Authentication', 'Currency Information', 'Product Information' (selected), 'Order Information', 'Tax Information', and 'Webhooks Configuration'. Under the 'Product Information' tab, there are two 'Fields and their uses' sections. The first section lists 'Connector Instance Name' (K\_dev) and 'Company' (My Company (San Francisco)). The second section lists 'Shopify Sync/Import Images?' (checked), 'Shopify Sync/Import Price?' (checked), and 'Stock Field Type' (Free To Use Quantity (product.product)).

## 2.3 Order Information

This page/tab displays all the information related to the import of orders.

- **Default Order Prefix:** Enable/Disable to give your customized Order Prefix for every Sale Order.
- **Order Prefix:** In case, you wish to disable the Default Order Prefix, this field lets you provide a prefix of your choice to be appended before every Sale Order. This will improve the visibility and identity of every Sale Order
- **Order Status:** Choose the Order of which status you wish to import from Shopify to Odoo. It can be Open, Paid, Completed, Pending, or all orders.
- **Import Order Through** We have given an option in Odoo Shopify Connector to import orders on the basis of Order status.
- **Warehouse:** Choose a warehouse, to which your stocks belong.
- **Sales Team:** Let's you choose a sales team for your sale orders.
- **Sales Person:** Represents the name of the concerning salesperson.

All the other details can be found in the 'Fields and their uses:-' section.

Shopify Connector Dashboard Instance Processes Reporting Configuration 4 3 My Company (San Francisco) Marc Demo

Shopify Instances / K\_dev

Save Discard 1 / 1 < >

Authentication Currency Information Product Information **Order Information** Tax Information Webhooks Configuration

**Fields and their uses :-**

- **Default Order Prefix** : Enables/disables – default order prefix of the Odoo
- **Order Prefix** : Prefix added on orders imported from shopify to odoo is shown here
- **Order Status** : Shows the configuration of the imported order status
- **Sales Team** : Shows the location of the sales team
- **Warehouse** : Shows the default warehouse for the stocks
- **Sales Person** : Displays the name of the sales person
- **Import Orders Through** : Shows the status of orders imported portal
- **Payment Terms** : Shows the duration of the payment required to get completed
- **Sale Workflow Configurations** : Shows the flow of the order when imported to the odoo
- **Emails Configurations** : Sends a monthly sales report to the email-ids provided in it.

Default Order Prefix

Order Prefix SC

Order Status

Import Orders through Status

Warehouse San Francisco

Sales Team Europe

Sales Person Mitchell Admin

Payment Term

If you will scroll down you will find two more configurations in the Order Information tab.

- **Sale Workflow Configurations**

You may use this feature to set the sale order status cycle for all orders imported from the instance. To add a line, click the 'Add a line' button and fill in the relevant information.

- **Email Configurations**

You can send the sale order report to the email addresses of your choice. Click on 'Add a line' to add an email address.

## Sale Workflow Configurations

Sale Workflow	Order Status
Add a line	

## Email Configurations

Add a line

## 2.4 Tax Information

You may apply tax to Shopify Sales Orders in two ways using this option.

- **Invoice TAX Account:** It displays the tax account which is utilized for the invoice tax default account.
- **Credit Note TAX Account:** It displays the tax account which will be used for the Credit Note/Refund tax default account.

**Shopify Connector** | Dashboard | Instance | Processes | Reporting | Configuration | My Company (San Francisco) | Marc Demo

Shopify Instances / K\_dev

Save | Discard | Deactivate Instance | Draft | Connected | Active | Deactivate

Price lists | Configure Schedulers | Active Schedulers

**Fields and their uses :-**

- Connector Instance Name : Displays Shopify Instance Name
- Company : Shows the name of the company

Connector Instance Name: K\_dev

Company: My Company (San Francisco)

Require Meta Mapping?

Authentication | Currency Information | Product Information | Order Information | **Tax Information** | Webhooks Configuration

**Fields and their uses :-**

- Invoice TAX Account : Show the tax account which will be used for invoice tax default account
- Credit Note TAX Account : Show the tax account which will be used for Credit Note/Refund tax default account

Invoice TAX Account: 101402 Bank | Credit Note TAX Account: 101501 Cash

101501 Cash

**ACTIVATED**

### **3. Webhook Configurations**

Webhooks are essentially a hook on Shopify, that produces an Odoo record as soon as the Shopify record is created/updated. It basically is the concept of real-time syncing of data from Shopify to Odoo

For Instance—if a merchant creates a new product or updates the existing one in Shopify, all the changes will be immediately synced with the Odoo.

It shows all of the webhooks that have been set up. The 8 webhook operations supported by our connector are: Order Create/Update, Product Create/Update, Customer Create/Update, and Collections Create/Update.

#### **3.1 How to set up a Webhook**

1. There are 8 possible types of operations in Webhooks, that are in-built and provided to a user.
2. When you select a certain webhook operation, it automatically captures the URL.
3. The Shopify Store ID gets assigned to it automatically, and the instance is attached to it by default whenever the user clicks on the “Sync” button present on it.
4. When a user clicks on the ‘Sync’ button for a particular Webhook, it will be activated. To deactivate the Webhook, click on the ‘Archive’ state that represents the Webhook is not active.

Shopify Connector Dashboard Instance Processes Reporting Configuration 4 3 My Company (San Francisco) Marc Demo

Shopify Instances / K\_dev Save Discard 1 / 1 < >

**K\_dev**

Company: My Company (San Francisco)  
 Require Meta Mapping?

Authentication Currency Information Product Information Order Information Tax Information **Webhooks Configuration**

**Fields and their uses :-**

- **Webhooks** : Webhooks are basically a hook on the shopify which creates a record on odoo as soon as the record is created on the shopify. Shows all the webhooks which are configured

Refresh

Name	Operation	Hook Status	Webhook Url	Shopify Id	Shopify Instance...	
Orders/Create	Order Create	Active	https://shopify14.kappso.com/shopify_hook/c2hvcGl...	10538054451...	K_dev	<span>Archive</span> <span>🗑</span>
Orders/Updated	Order Update	Delete	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>
Products/Create	Product Create	Delete	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>
Products/Update	Product Update	Delete	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>
Customers/Create	Customer Create	Active	https://shopify14.kappso.com/shopify_hook/c2hvcGl...	10544482878...	K_dev	<span>Archive</span> <span>🗑</span>
Customers/Update	Customer Update	Disabled	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>
Collections/Create	Collections Create	Disabled	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>
Collections/Update	Collections Update	Disabled	https://shopify14.kappso.com/shopify_hook/c2hvcGl...		K_dev	<span>Sync</span> <span>🗑</span>

Here, Order Create is in an Active state or Sync State. It means whenever a new order is created at the Shopify end, it will sync all the data related to that Order at the Odoo end as well.

Depending on your preferences, you can activate and deactivate a Webhook.

**Note:** It is worth noting that when an instance is disabled, the webhooks associated with it are immediately set to 'disabled.' When an instance is revived, the associated webhooks are not automatically triggered and must be activated manually.

We have also given a 'Refresh' button so that you can instantly sync Shopify activities with Odoo.

#### 4. Odoo Shopify Operations

After successful configurations, it's time to perform the vital Shopify Odoo Integration. Now you can easily perform various import and export operations from Odoo without using the Shopify backend.

Let's see further how import and export functions are performed between Odoo and Shopify.

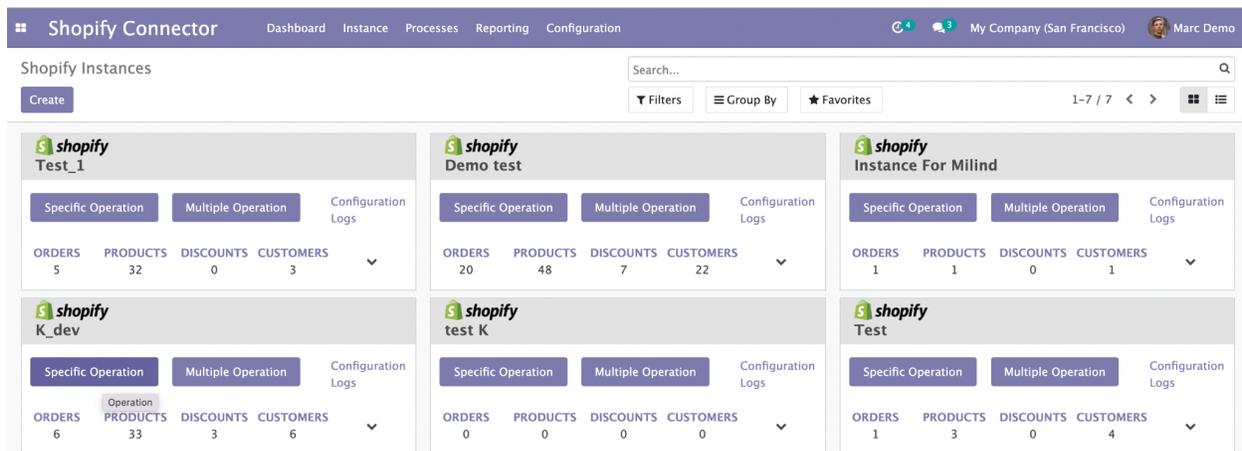
## 4.1 Import Operations

Import operations are performed in two ways: i) Importing records in bulk ii) Importing records using the Specific Id.

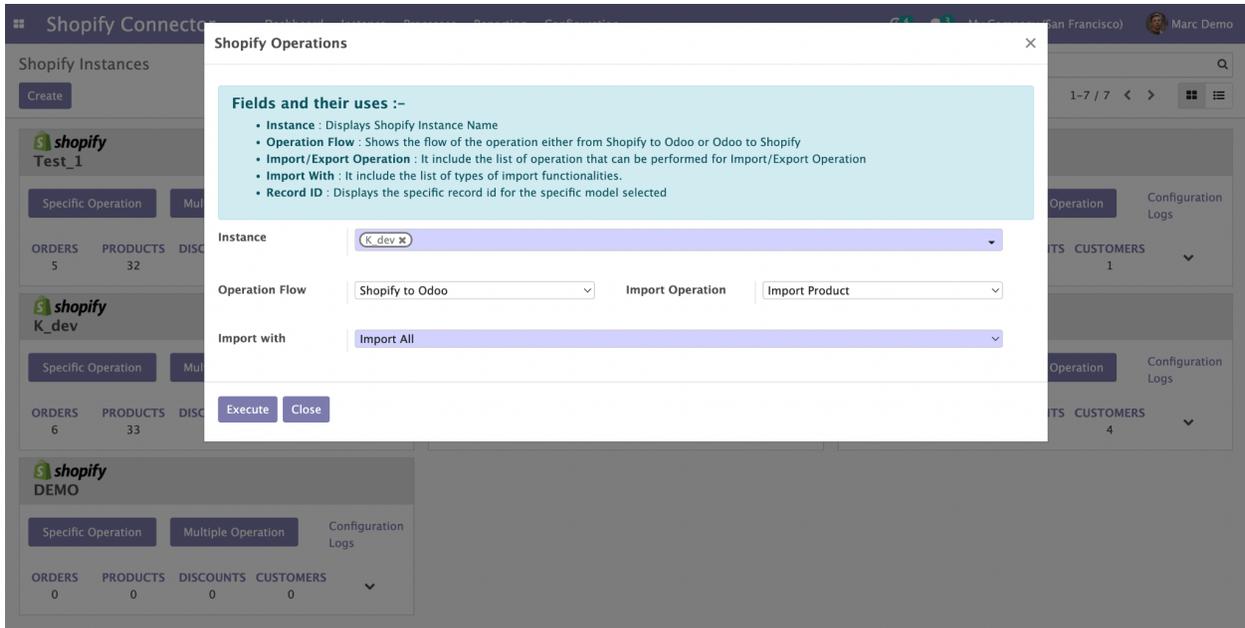
### a. Import Products

#### i) Using 'Import All'

Go to the Shopify Connector in Odoo. Choose your desired Instance for performing import operations. Click on the '**Specific Operation**' button.

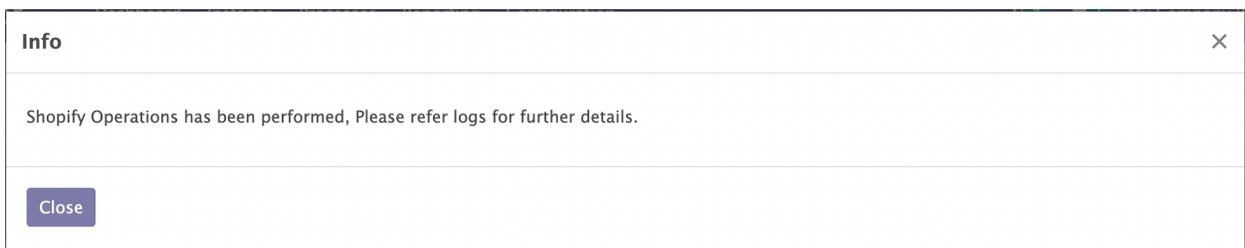


Here, we are choosing the 'K\_dev' Instance. On clicking the specific operation button, a window will appear as given below:



- **Instance:** Choose the instance id you wish to perform import operations with.
- **Operation Flow:** Select 'Shopify to Odoo' while importing.
- **Import Operation:** Select 'Import Product' from various given operations.
- **Import With:** Select 'Import All'.

After selecting the required entries, click on the 'Execute' button. A display box will appear showing that the import operations have been performed successfully.



Now to verify the import operation in progress, click on Processes>Queue Jobs. Here you will find the records with New, In-Progress Successful, and Failed status.

Shopify Connector							
Dashboard Instance Processes Reporting Configuration							
All Queue Jobs							
Name	Shopify ID	Operation Flow	Operation Type	Shopify Instance	State		
<input type="checkbox"/>	Vintage Sweatshirt in Light Grey	6724630511673	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	U1002	6726927941689	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	U1001	6726924075065	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Tshirts	6724630642745	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Tshirts	6724630577209	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Tshirts	6724630544441	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	tshirt003	6724630413369	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	tshirt002	6724630446137	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	tshirt001	6724630478905	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Tshirt White	6724627136569	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Tshirt	6707438288953	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	test009	6695982563385	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Symphonie Sweatshirt in Black	6724630675513	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Steel Bottles	6729172713529	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Stands	6726885572665	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Rin 003	6696193589305	Product Template	Shopify to Shopify Layer	Import	K_dev	Done

If you wish to see your newly imported products, click on the 'State' tab. Once you click on it, you will see your imported products at the top.

[This shows your new imported products]

Shopify Connector							
Dashboard Instance Processes Reporting Configuration							
All Queue Jobs							
Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State	
<input type="checkbox"/>	chair123	6707019350073	Product Template	Shopify to Shopify Layer	Import	K_dev	In Progress
<input type="checkbox"/>	Apple	6697934585913	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	bags	6706958008377	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	1_image_test_01	6695875903545	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	5A	6695875838009	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 001	6702086979641	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	6-A	6695875772473	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 002	6702148452409	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 003	6702446149689	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 003	6702450245689	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 004	6702460239929	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 004	6702460862521	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-1	6693668913209	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	A-Poco 004	6702461681721	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	1_image_test	6695876034617	Product Template	Shopify to Shopify Layer	Import	K_dev	New

You can compare your imported products from Shopify as well. Refer to the image below:

The screenshot shows the Shopify admin interface for a store named 'ksolves-dev-store...'. The 'Products' section is active, showing a list of products. The table below represents the data visible in the screenshot:

Product	Status	Inventory	Type	Vendor
1_image_test	Active	7 in stock for 2 variants		ksolves-dev-store-001
1_image_test_01	Active	0 in stock for 2 variants		ksolves-dev-store-001
5A	Active	0 in stock for 2 variants		ksolves-dev-store-001
6-A	Active	0 in stock for 2 variants		ksolves-dev-store-001
A-1	Active	89 in stock		ksolves-dev-store-001
A-Poco 001	Active	450 in stock		ksolves-dev-store-001
A-Poco 002	Active	0 in stock for 2 variants		ksolves-dev-store-001
A-Poco 003	Active	0 in stock for 2 variants		ksolves-dev-store-001

The Import Product Operation is successful!

## ii) Import Using 'Specific ID'

You can import an already created product in Shopify or you can create a new product and then import it using its Specific ID.

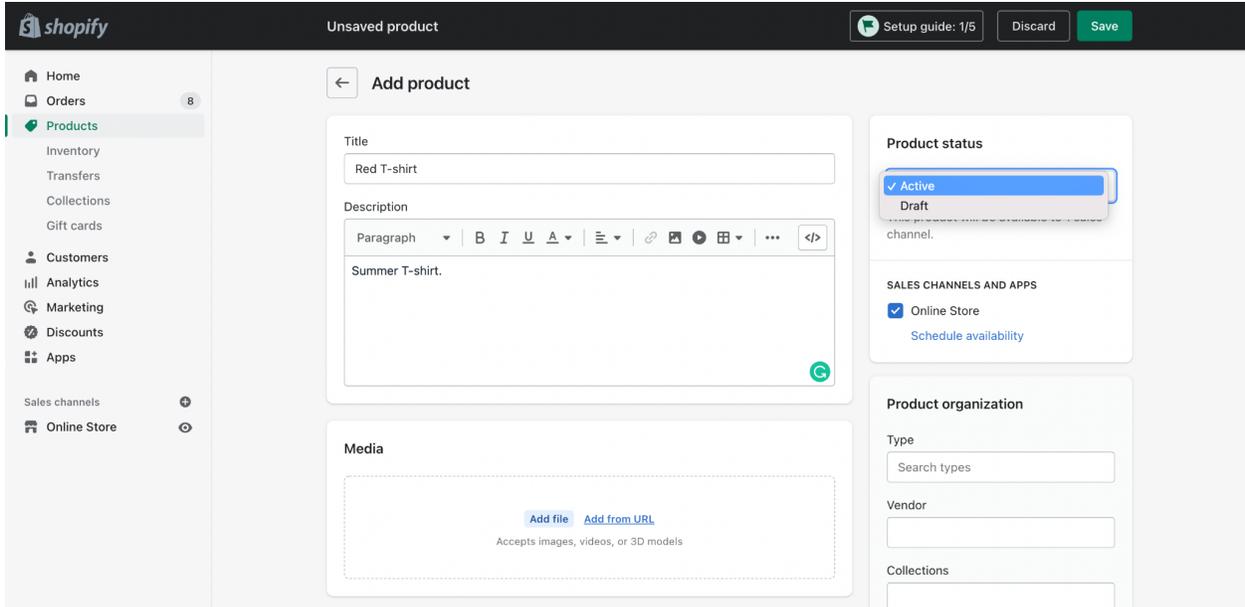
Let's take an example by creating a new product in Shopify.

### A.1 Create Product in Shopify

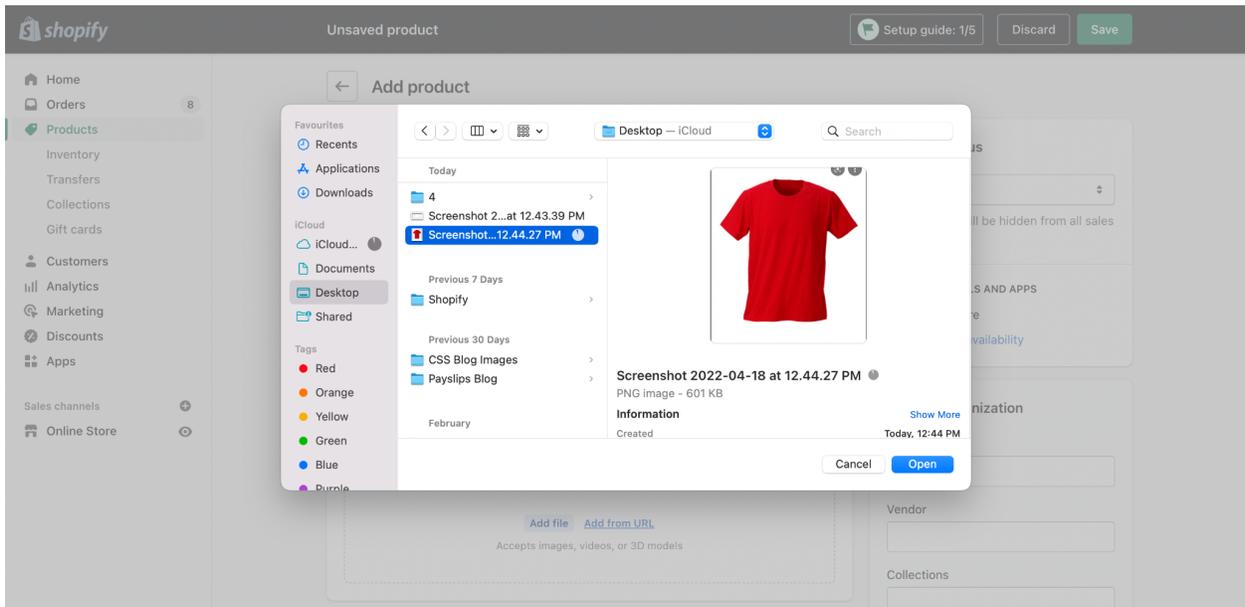
Creating a New Product in Shopify: Go to Shopify backend > Go to Products > Click on the 'Add Product' button.

A window will appear where you can fill all the relatable fields about your new product. Here, we have taken an example of a 'Red T-shirt' as a new product.

Give product title, and description, and change the product status to 'Active'.



Now, click on 'Add File' in the Media section to give some interactive images to your product.



After selecting the image, scroll down to Pricing, Inventory section, and Quantity section.

Select the Price, SKU, Bar Code, and quantity. You can fill in as many details as you like about your product.

The screenshot shows the 'Unsaved product' page in Shopify. The left sidebar contains navigation options: Home, Orders (8), Products (selected), Inventory, Transfers, Collections, Gift cards, Customers, Analytics, Marketing, Discounts, Apps, Sales channels, and Online Store. The main content area is divided into three sections:

- Pricing:** Price is ₹ 1,000.00, Compare at price is ₹ 0.00. There is a checkbox for 'Charge tax on this product' which is checked. Cost per item is ₹ 0.00.
- Inventory:** SKU (Stock Keeping Unit) is 'Tee-1', Barcode (ISBN, UPC, GTIN, etc.) is '12345'. There is a checkbox for 'Track quantity' which is checked, and a checkbox for 'Continue selling when out of stock' which is unchecked.
- QUANTITY:** Location name is 'dlf phase 3', and the quantity is '500'.

At the top right, there are buttons for 'Setup guide: 1/5', 'Discard', and 'Save'.

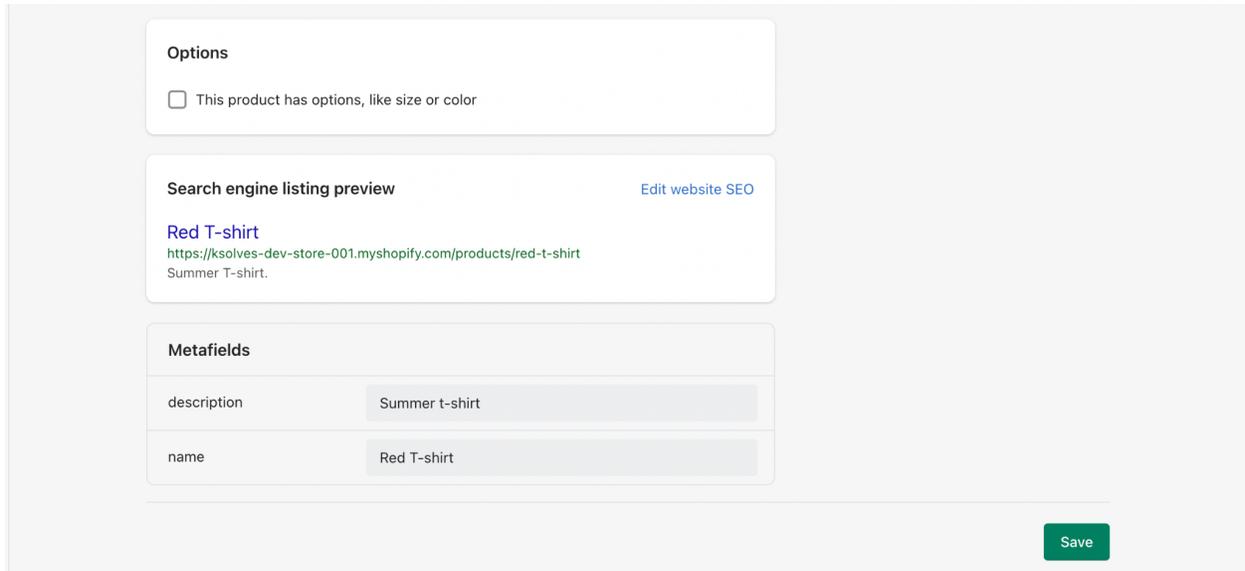
Scroll down the page and mention the weight of your product.

The screenshot shows the 'Unsaved product' page in Shopify, scrolled down to the 'Shipping' section. The left sidebar is the same as in the previous screenshot. The main content area shows:

- Shipping:** A checkbox for 'This is a physical product' is checked.
- WEIGHT:** Used to calculate shipping rates at checkout and label prices during fulfillment. The weight is '0.5' kg.
- CUSTOMS INFORMATION:** Customs authorities use this information to calculate duties when shipping internationally. Shown on printed customs forms.
- Country/Region of origin:** A dropdown menu with 'Select country/region' selected. Below it, it says 'In most cases, where the product is manufactured.'
- HS (Harmonized System) code:** A search box with 'Search or enter a HS code' and a note: 'Manually enter codes that are longer than 6 numbers.'

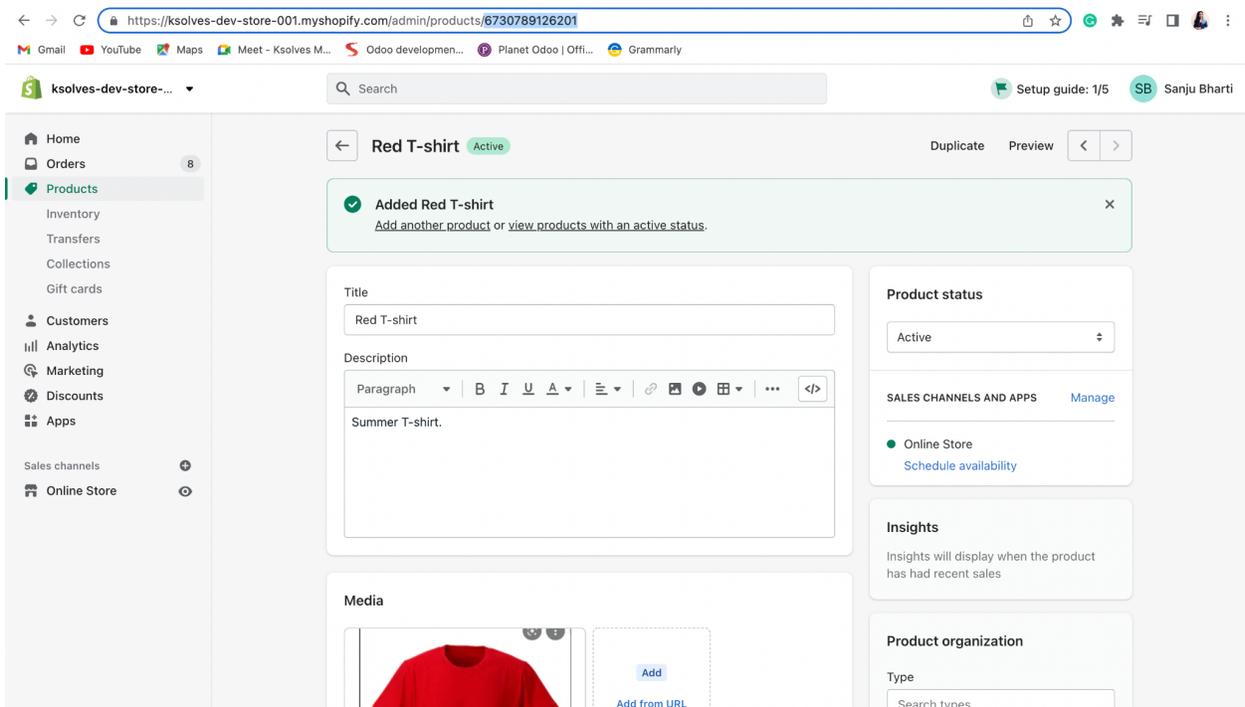
At the top right, there are buttons for 'Setup guide: 1/5', 'Discard', and 'Save'.

After giving all the details, click on the 'Save' button available at the bottom right corner.



As your new product is created in Shopify. Go to Shopify>Products>Select the newly created product>And copy its unique Id as shown below:

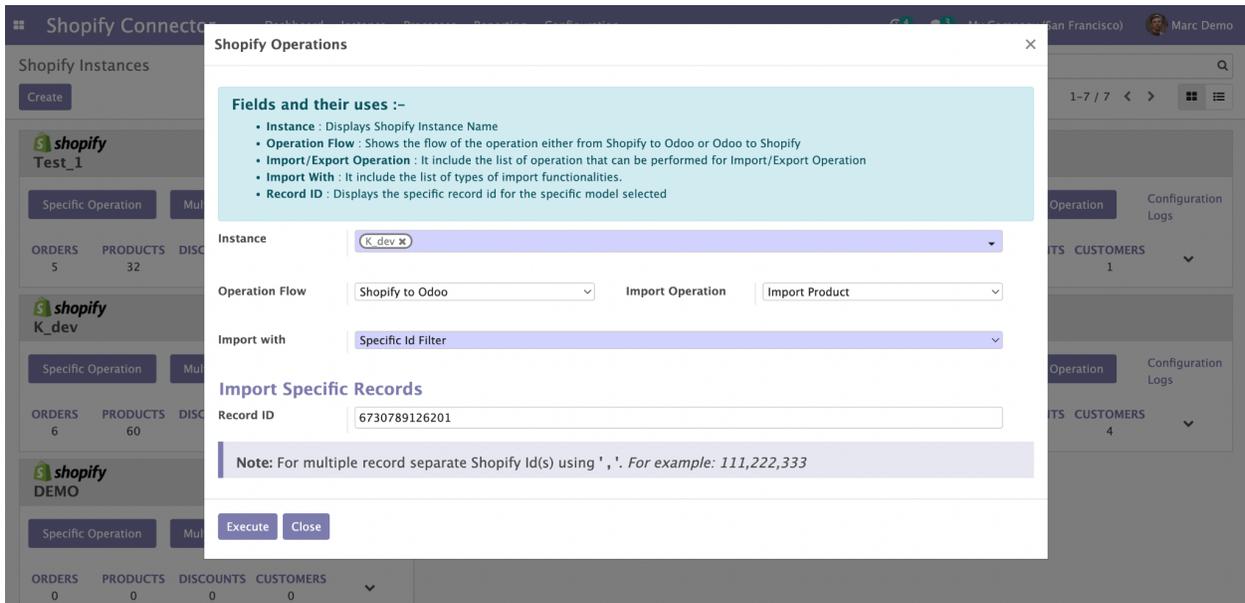
The unique Id here is:6730789126201



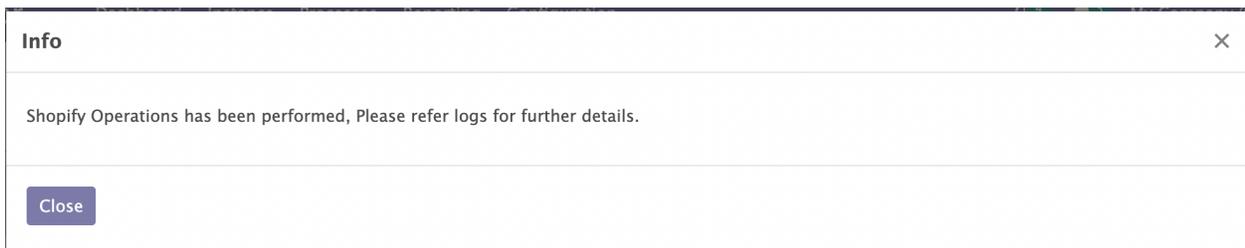
Go to Odoo Shopify Connector>Go to Instance>Choose Specific Operation

- **Import With:** Specific Id Filter

- **Record Id:** Paste the copied Id of your new product, and click on the 'Execute' button.



A display box will appear showing that the import operations have been performed successfully.



To quickly check your product to be imported, Go to Processes>Queue Jobs. Here you will find your product to be imported.

Shopify Connector							
Dashboard		Instance	Processes	Reporting	Configuration	My Company (San Francisco) Marc Demo	
All Queue Jobs		Search...		1-68 / 68			
<input type="checkbox"/> Name		Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	K_dev	New
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	Test	Done
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/>	Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	Test	Done
<input type="checkbox"/>	Vintage Sweatshirt in Light Grey	6724630511673	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	U1002	6726927941689	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	U1001	6726924075065	Product Template	Shopify to Shopify Layer	Import	K_dev	Done

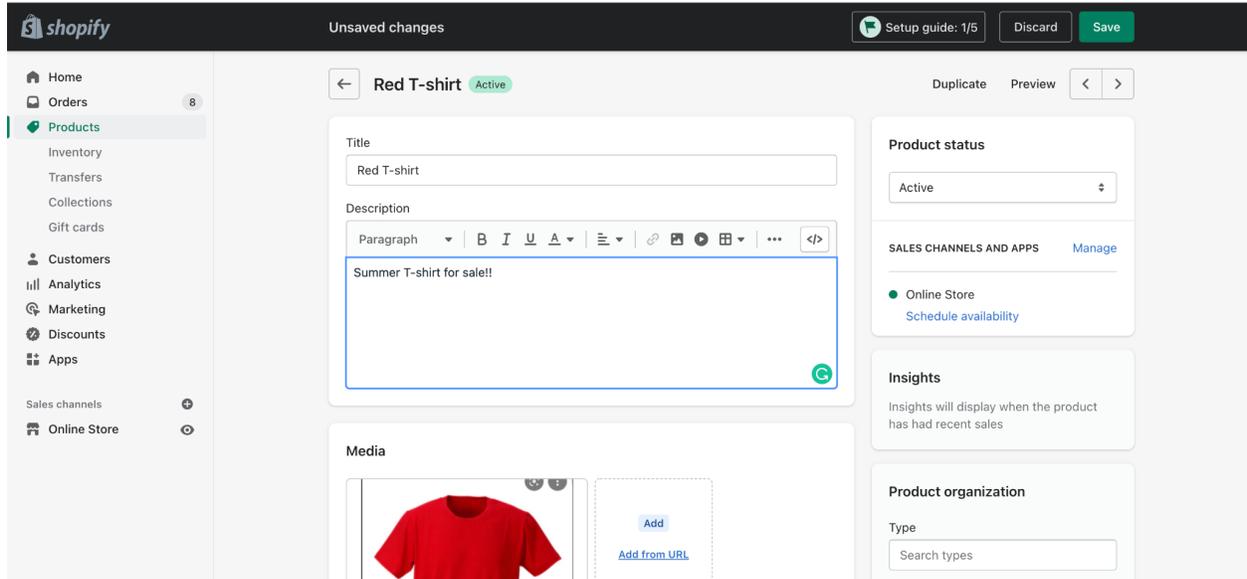
## A.2 Update Product in Shopify

After importing a product from Shopify to Odoo, if any changes are made in the product from the Shopify end, it will get synced to the Odoo end as well. How? Let's see.

**Note:** In section 3, we discussed the Webhook activation and deactivation. If the Product Update webhook is activated, the changes made in the Product will get updated automatically.

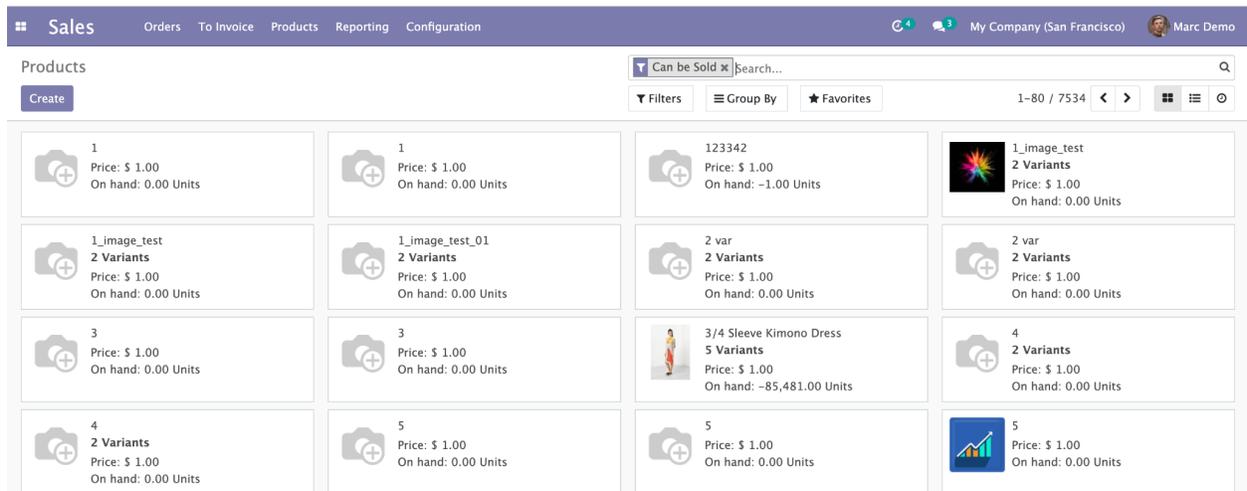
Go to Shopify>Open a Product>Edit the product's description (or anything as per your choice). Click on the 'Save' button.

For Example: Here we have updated the description of our Red T-shirt product from 'Summer T-shirt' to 'Summer T-shirt for sale!!'

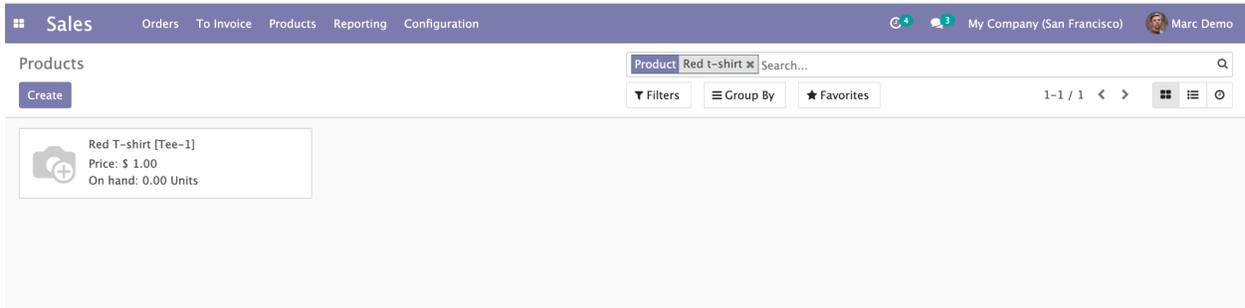


Once the product details are updated, go to Odoo>Sales>Products>Products. Search your recently updated product in the search bar.

For example Red T-shirt

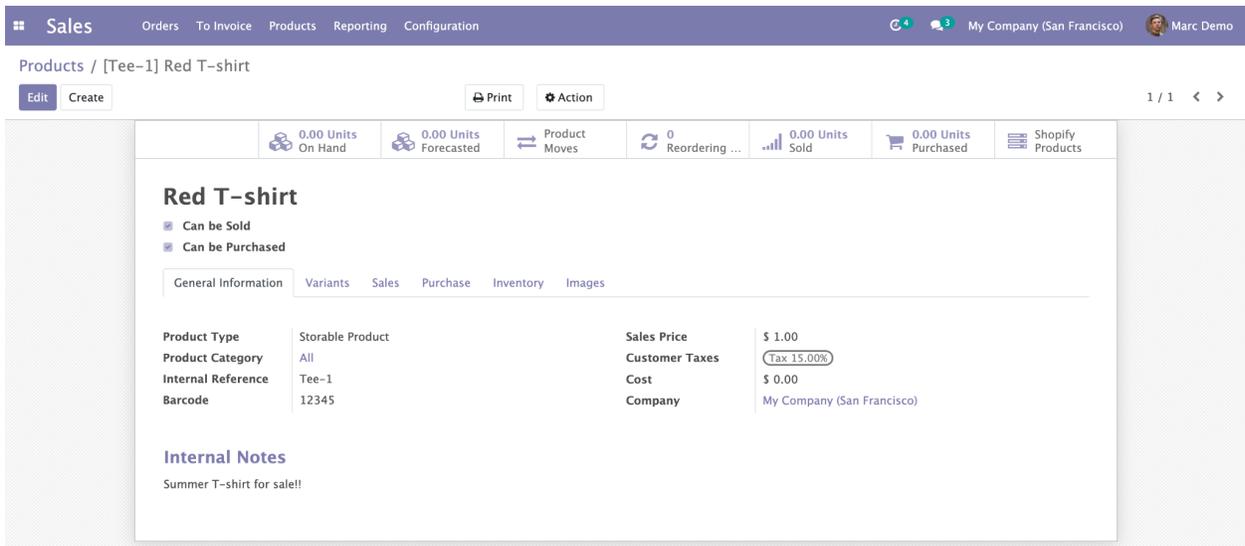


Find your product and open it.



You will see that the description of the product has been updated automatically.

It happens as our 'Product Update' Webhook is activated.



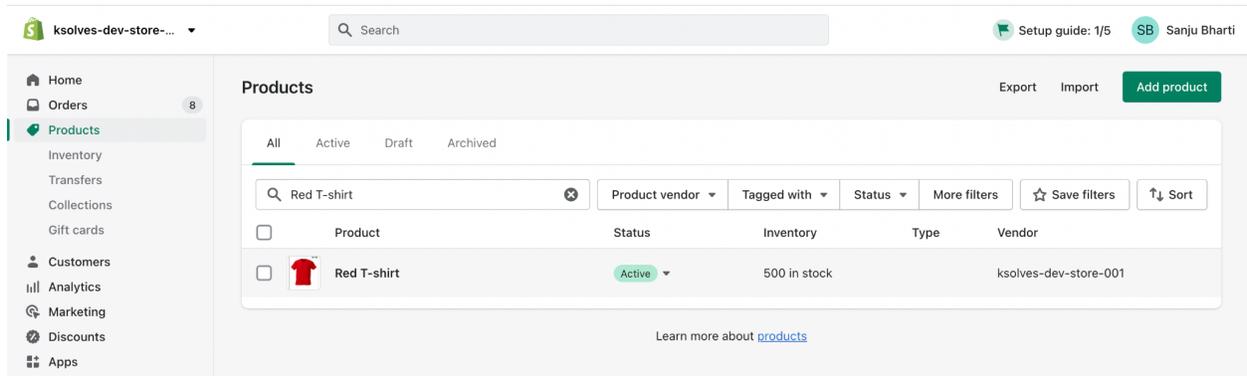
Hence, this way your product will get updated on both ends.

## b. Import Product Stock

Importing stock is necessary as it shows how much quantity you have in the inventory for sale.

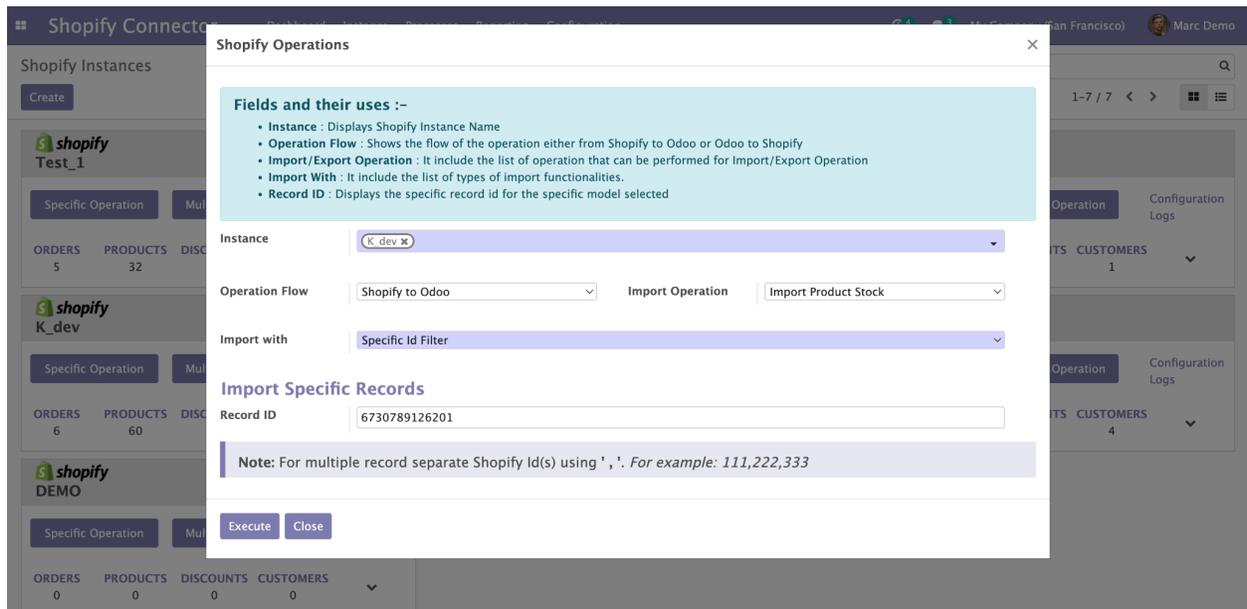
From the above section A.2, we can see that our product Red T-shirt is imported and updated with all the description parts. However, the 'On Hand units' or stock is not updated.

Remember, the quantity we have given to the Red t-shirt product is 500. Let's import the product stock now.



**Note:** In order to import the stock, the product must be imported first.

Go to Instance>Specific Operation>Import Product Stock>Use the Specific ID of the product and click on the 'Execute' button.



A window will appear showing the successful import stock operation.

**Info** ✕

Shopify Operations has been performed, Please refer logs for further details.

[Close](#)

Go to Processes>Queue Jobs. The Inventory Adjustment shows that the stock has been synced between Shopify and Odoo.

Shopify Connector						
All Queue Jobs						
Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/> Inventory Adjustment	0	stock	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Red T-shirt	6730789126201	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> 1_image_test	6695876034617	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> 1_image_test	6695876034617	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Vintage Sweatshirt in Light Grey	6724630511673	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> U1002	6726927941689	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> U1001	6726924075065	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> Tshirts	6724630642745	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> Tshirts	6724630577209	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> Tshirts	6724630544441	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> tshirt003	6724630413369	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> tshirt002	6724630446137	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> tshirt001	6724630478905	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> Tshirt White	6724627136569	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> Tshirt	6707438288953	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done
<input type="checkbox"/> test009	6695982563385	Product Template	Shopify to Shopify Layer	Import	K_dev 1	Done

Now, go to Inventory from App Menu>Operations>Inventory Adjustments

**Inventory** | Overview | **Operations** | Products | Reporting | Configuration

Inventory Overview

Receipts: 6 To Process, 6 Late

Orders: 9 To Process, 4 Waiting, 13 Late, 2 Back Orders

Transfers

Replenishment

**Inventory Adjustments**

Scrap

Run Scheduler

Here, you will find that your inventory\_adjustment is In Process.

Inventory Date	Inventory Reference	Company	Status
<input type="checkbox"/> 04/18/2022 14:39:35	product_inventory_2022-04-18 09:09:35	My Company (San Francisco)	In Progress
<input type="checkbox"/> 04/18/2022 14:10:43	Inventory	My Company (San Francisco)	Validated
<input type="checkbox"/> 04/14/2022 12:41:37	product_inventory_2022-04-14 07:11:19	My Company (San Francisco)	Validated
<input type="checkbox"/> 04/13/2022 12:28:03	product_inventory_2022-04-13 06:58:03	My Company (San Francisco)	In Progress
<input type="checkbox"/> 04/13/2022 12:00:53	product_inventory_2022-04-13 06:30:17	My Company (San Francisco)	Validated
<input type="checkbox"/> 04/13/2022 10:37:59	product_inventory_2022-04-13 05:07:40	My Company (San Francisco)	Validated
<input type="checkbox"/> 01/31/2022 20:07:57	product_inventory_2022-01-31 14:37:57	My Company (San Francisco)	In Progress
<input type="checkbox"/> 01/31/2022 20:05:53	product_inventory_2022-01-31 14:35:53	My Company (San Francisco)	In Progress
<input type="checkbox"/> 01/31/2022 19:27:45	product_inventory_2022-01-31 13:57:45	My Company (San Francisco)	In Progress
<input type="checkbox"/> 01/31/2022 19:26:55	product_inventory_2022-01-31 13:56:55	My Company (San Francisco)	In Progress
<input type="checkbox"/> 01/04/2022 23:04:06	product_inventory_2022-01-04 17:34:06	My Company (San Francisco)	In Progress
<input type="checkbox"/> 10/06/2021 14:50:36	product_inventory_2021-10-06 09:19:07	My Company (San Francisco)	Validated
<input type="checkbox"/> 08/27/2021 11:55:51	product_inventory_2021-08-27 06:25:51	My Company (San Francisco)	In Progress

Click and open the recently created inventory\_adjustment. A window containing all the details will be displayed.

Inventory Adjustments / product\_inventory\_2022-04-18 09:09:35

1 / 39 < >

**product\_inventory\_2022-04-18 09:09:35**

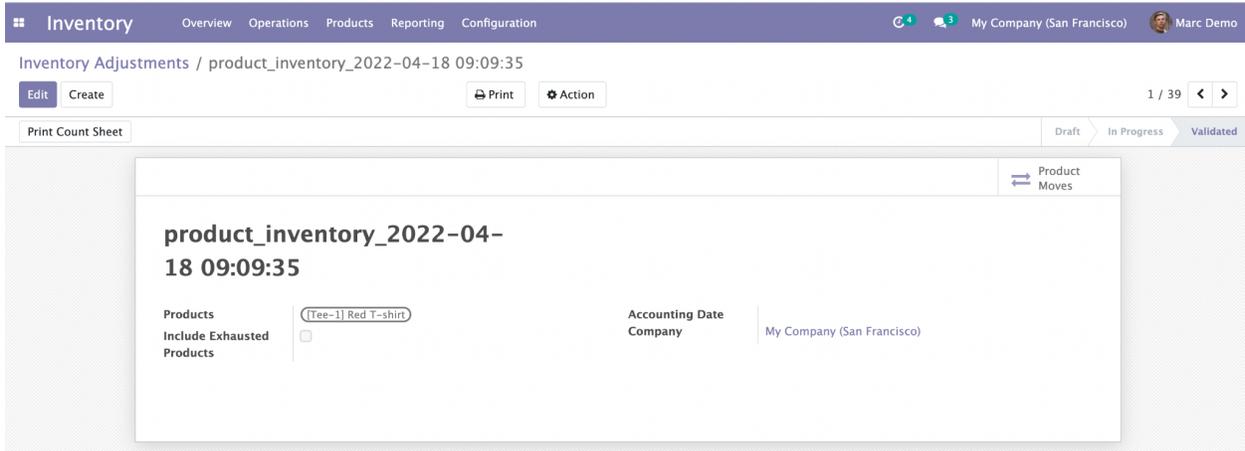
**Products**

**Include Exhausted Products**

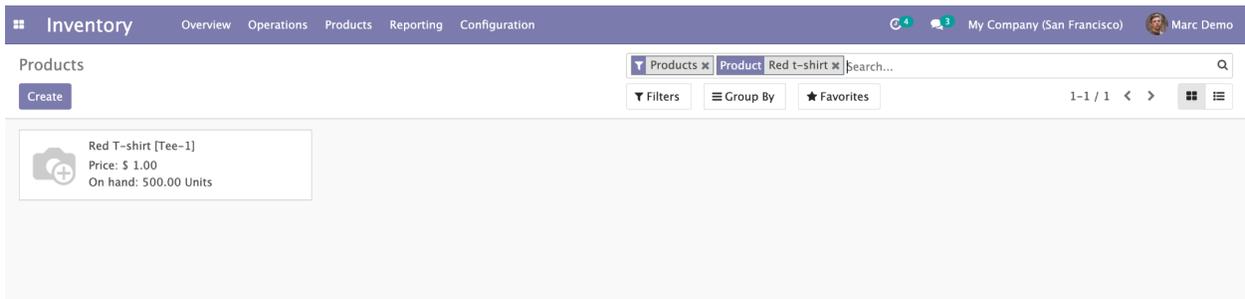
**Accounting Date**

**Company** My Company (San Francisco)

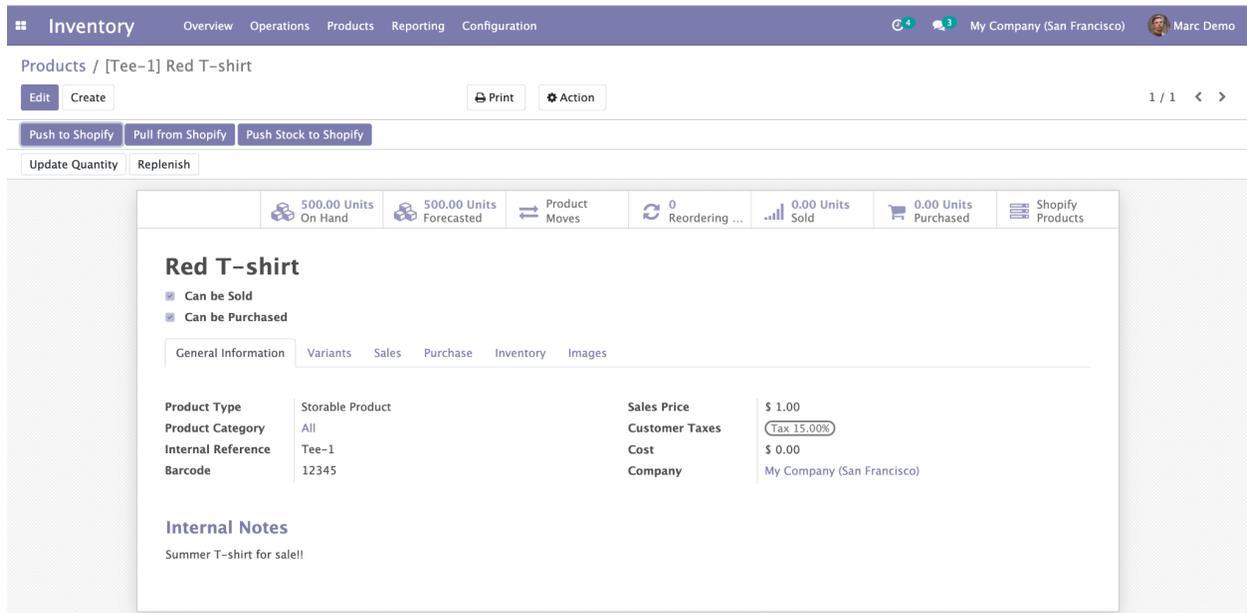
Now, we have kept this process manual. Click on the 'Validate Inventory' to update the product stock.



Go to Products>Products and search for your product. For example Red T-shirt



Open your product and you will find that the On Hand units are updated from 0 to 500.



This is how the product stock is imported from Shopify to Odoo.

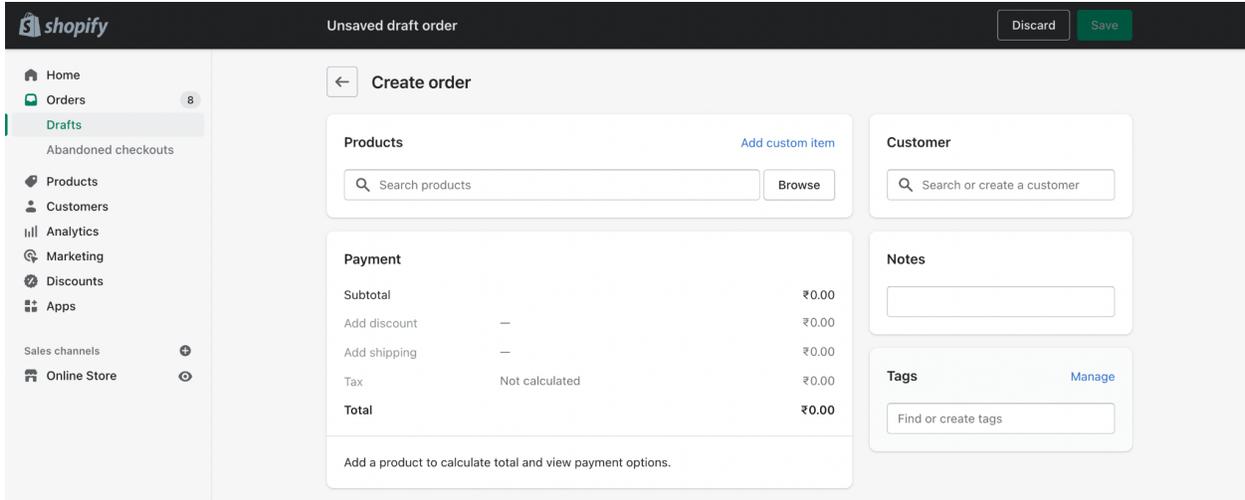
Things to Remember:

- Stocks are only imported if the products associated with them are already imported.
- Only storable products will have the On Hand units, unstorable products do not have On Hand units (quantity).

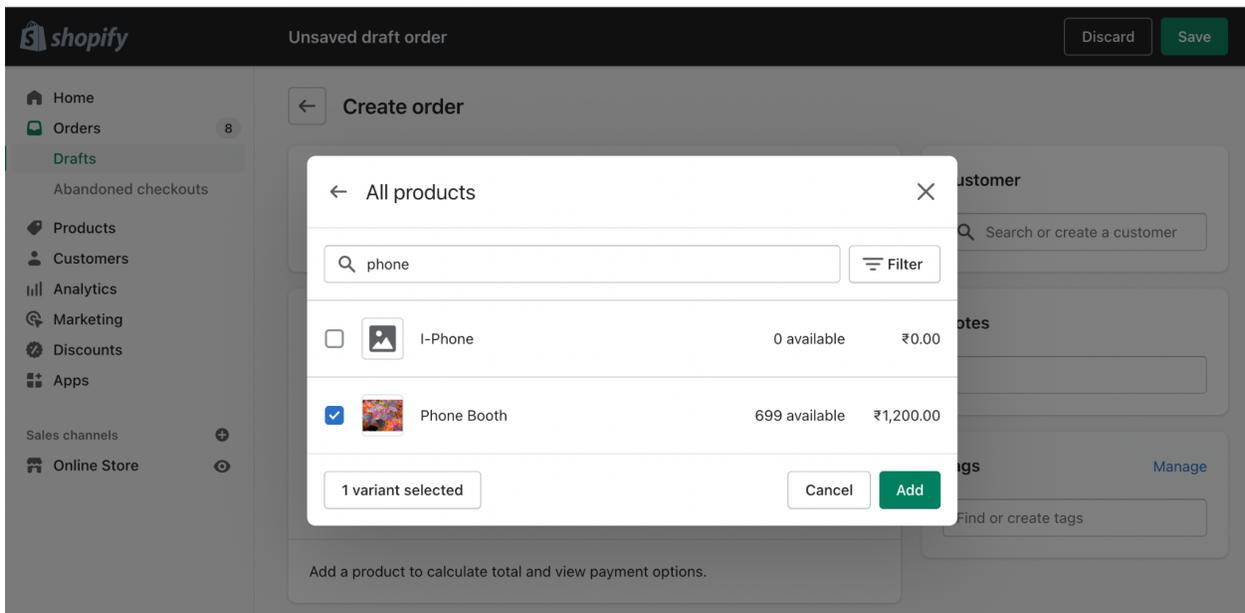
### c. Import Order

You can import the old orders or can create a new one and perform the import function.

Go to Shopify backend>Orders>Click on the 'Create Order' button.

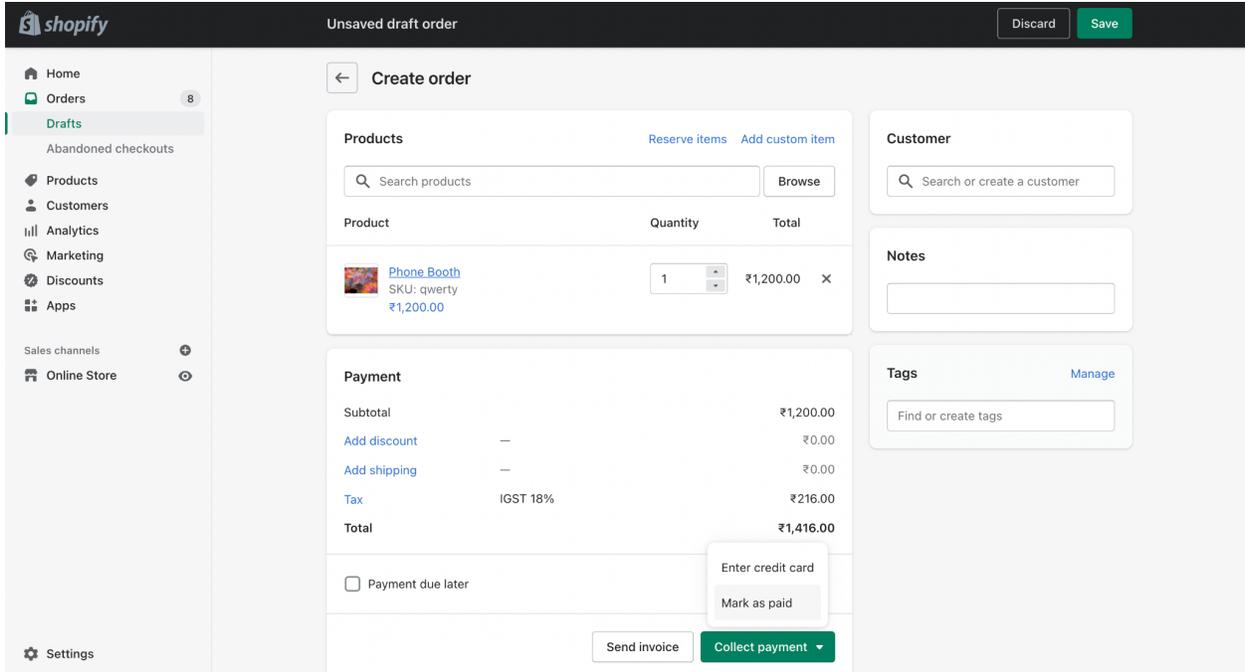


Search for the product, for example, Phone Booth, and click on Add.

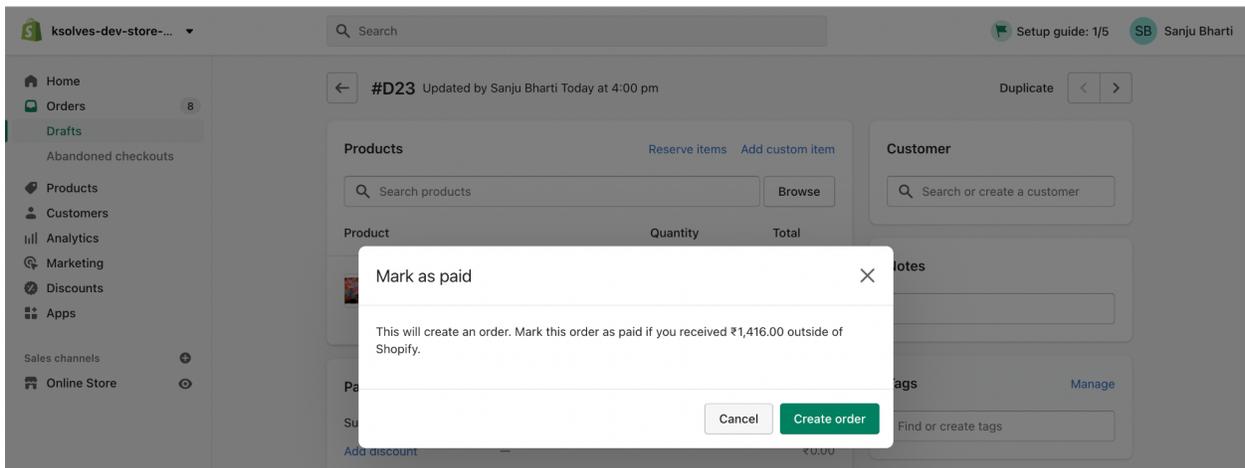


Once you click on Add, you will see that your product has been added. Now if you scroll down, at the bottom, you will find the 'Collect Payment' option.

If you wish you can save your order as it is. However, if you want to import a fulfilled order, select 'Mark as paid' as shown below:



A dialogue box will appear that will ask you to 'Cancel' or 'Create Order'. Choose the create order option.



Now as you can see your order status is paid and unfulfilled. Click on the 'fulfill item' button.

ksolves-dev-store-... Search Setup guide: 1/5 SB Sanju Bharti

Home Orders 9 Drafts Abandoned checkouts Products Customers Analytics Marketing Discounts Apps Sales channels Online Store

#1022 Paid Unfulfilled 18 April 2022 at 4:00 pm from Draft Orders Refund Edit More actions

**Unfulfilled (1)**

1 Phone Booth ₹1,200.00 × 1 ₹1,200.00 SKU: qwerty

Fulfill item

**Notes** Edit  
No notes from customer

**Customer**  
Search or create a customer

**CONTACT INFORMATION** Edit  
No email provided  
No phone number

**SHIPPING ADDRESS** Edit  
No shipping address provided

**Paid**

Subtotal	1 item	₹1,200.00
Tax	IGST 18%	₹216.00
<b>Total</b>		<b>₹1,416.00</b>
Paid by customer		₹1,416.00

Now the order is in the unfulfilled state only. Again click on the ‘Fulfil Item’ button.

ksolves-dev-store-... Search Setup guide: 1/5 SB Sanju Bharti

Home Orders 9 Drafts Abandoned checkouts Products Customers Analytics Marketing Discounts Apps Sales channels Online Store

Fulfill item Print packing slip

#1022 Unfulfilled dlf phase 3

**ITEMS**

Phone Booth 12 kg 1 of 1 SKU: qwerty

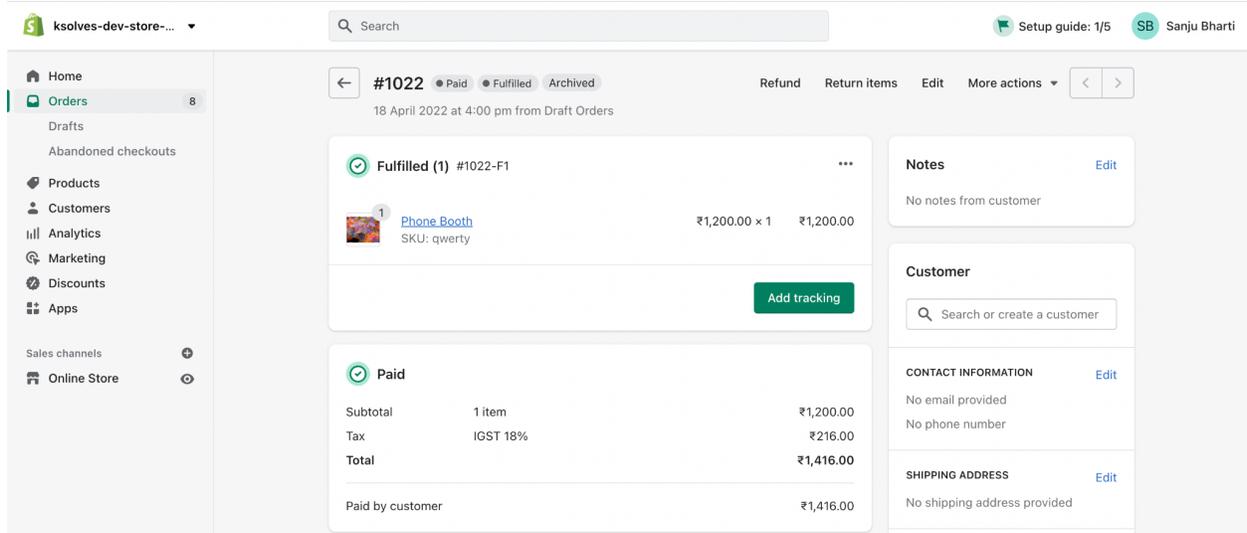
**Shipping address** Edit  
This order doesn't have a shipping address.

**Summary**  
Fulfilling from dlf phase 3  
1 of 1 item

Fulfill item

Learn more about [fulfilling orders](#).

Now, finally, your order will be Paid, Fulfilled, and Archived.



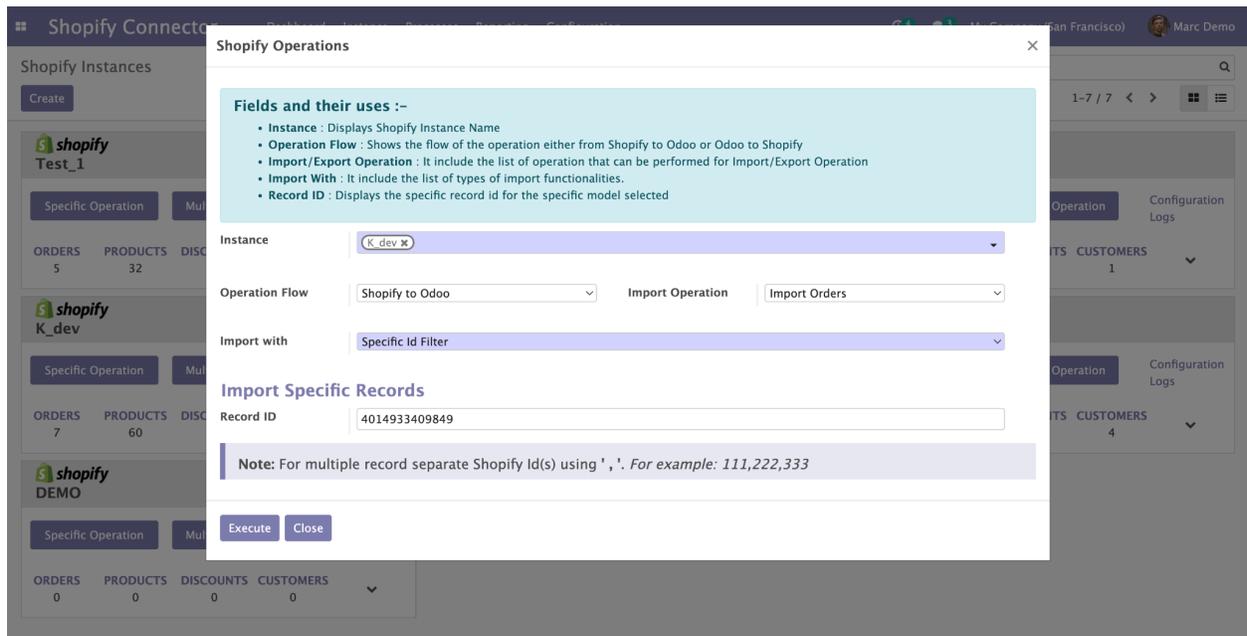
Select the unique Id of the order. Here the Order ID is 4014933409849.

Go to Shopify Connector>Instance>Specific Operation>Import Order

**Import with:** Specific id filter

**Record ID:** 4014933409849

Click on the Execute button.



A window displaying successful execution will appear.

**Info** ✕

Shopify Operations has been performed, Please refer logs for further details.

[Close](#)

Go to Processes>Queue Jobs. The Order to be Imported is in the New state.

Shopify Connector						
All Queue Jobs						
Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/> Sale Order #4014933409849	4014933409849	Sale Order	Shopify To Odoo	Import	K_dev	New
<input type="checkbox"/> Sale Order #4014933409849	4014933409849	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4014933409849	4014933409849	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4014933409849	4014933409849	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4014933409849	4014933409849	Sale Order	Shopify To Odoo	Import	K_dev	Done

You can wait for the cron to import the order, or you can do it manually.

Click on the order, and a window will appear containing all the required details of the order.

Click on the 'Process Record' button to complete the import operation manually.

**Shopify Connector** | Dashboard | Instance | Processes | Reporting | Configuration | My Company (San Francisco) | Marc Demo

All Queue Jobs / Sale Order #4014933409849

Action 1 / 80

Process Record
New | In Progress | Done | Failed

**Sale Order #4014933409849**

<b>Instance Information</b>		<b>Other Information</b>	
Domain	Sale Order	Operation Flow	Shopify To Odoo
Shopify ID	4014933409849	Operation Performed	
Odoo ID	0	Operation Type	Import
Shopify Instance	K_dev	Created on	04/18/2022 16:22:19

[Data](#)

As soon as you click on the process record button, the order will be processed successfully, and as shown the enqueued record will be in the Done state now.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo

All Queue Jobs / Sale Order #4014933409849

Action 1 / 80

New In Progress Done Failed

### Sale Order #4014933409849

Instance Information		Other Information	
Domain	Sale Order	Operation Flow	Shopify To Odoo
Shopify ID	4014933409849	Operation Performed	
Odoo ID	208	Operation Type	Import
Shopify Instance	K_dev	Created on	04/18/2022 16:22:19

Data

You can check the imported orders from sales as well.

Go to Dashboard>Orders. Here you will find that your order is created under a Guest customer as we did not give any customer details while creating an order. Therefore, it will automatically be imported under the Guest Customer.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo

Dashboard

Shopify Instance x Search...

Filters Group By Favorites 1 - 1 / 1

### Bar Chart

Legend: New State (Red), Progress State (Orange), Completed State (Yellow), Failed State (Cyan)

Category	New State	Progress State	Completed State	Failed State
Products	0	0	130	10
Collections	0	0	0	0
Coupons	0	0	0	0
Customers	0	0	0	0
Orders	0	0	5	5

7 Shopify Instances

145 Products	198 Variants
90 Attributes	27 Collections
10 Discounts	36 Customers
34 Orders	167 Delivery
164 Invoices	5 Refunds
4 Gateways	35 Inventory

Number	Order Date	Customer	Salesperson	Instance	Shopify Id	Next Activity	Company	Total	Shopify Status	Sync Stat...	Invoice Status
SC #401493340...	04/18/2022	Guest Customer	Mitchell Admin	K_dev	4014933409849		My Company (San...	₹ 1,416.00	Completed		Nothing to Invoice
SC #397484500...	04/15/2022	Guest Customer	Joel Willis	K_dev	3974845005881		My Company (San...	\$ 23.60	Completed		Nothing to Invoice
SC #401092303...	04/14/2022	A.D.Motors Spa	Mitchell Admin	K_dev	4010923032633		My Company (San...	₹ 120,000.00	Completed		To Invoice
SC #401017176...	04/14/2022	Anshu Anshu	Mitchell Admin	K_dev	4010171760697		My Company (San...	₹ 1,416.00	Paid		To Invoice
SC #400992364...	04/14/2022	Sukriti chatterjee	Mitchell Admin	K_dev	4009923641401		My Company (San...	₹ 200.00	Paid		Fully Invoiced
SC #108667034...	04/14/2022	gomez	Marc Demo	Instance For Milind	1086670340321		My Company (San...	\$ 100.00	Open		Nothing to Invoice
SC #862780424...	04/13/2022	A.D.Motors Spa	Mitchell Admin	K_dev	862780424249		My Company (San...	\$ 1,200.00	Open		Fully Invoiced
S00043	04/13/2022	Sukriti chatterjee	Mitchell Admin	K_dev	4003692249145		My Company (San...	₹ 1.15	Completed		Fully Invoiced
SC #400369025...	04/13/2022	Guest Customer	Mitchell Admin	K_dev	4003690250297		My Company (San...	\$ 1,416.00	Paid		Fully Invoiced
SC #398465499...	01/31/2022	K a s h i s h Kashish	Joel Willis	Demo test	3984654991404		My Company (San...	₹ 171.10	Pending		Fully Invoiced
SC #398399763...	01/31/2022	kashish ahuja	Joel Willis	Demo test	3983997632556		My Company (San...	₹ 59.00	Completed		Fully Invoiced
SC #398327624...	01/31/2022	d e h r a m	Joel Willis	Demo test	3983276245036		My Company (San...	₹ 29.50	Completed		Fully Invoiced
SC #398325756...	01/31/2022	kashish ahuja	Joel Willis	Demo test	3983257567276		My Company (San...	₹ 88.50	Partially Paid		Fully Invoiced

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo

Dashboard / Shopify Quotation / SC #4014933409849

Edit Create Print Action 1 / 34

Send by Email Confirm Cancel Quotation Quotation Sent Sales Order

Customer Preview Synchronised With Shopify

**SC #4014933409849**

**Customer** Guest Customer  
patna  
Apartment  
patna 800001  
Bihar BR  
India

**Expiration**  
Payment Terms  
Auto Workflow  
Process

KS: Automatic Validation

**Invoice Address** Guest Customer  
**Delivery Address** Guest Customer, Veronica mignonne  
**Quotation Template**

Order Lines Optional Products Other Info Shopify Info

Product	Description	Quantity	Unit Price	Taxes	Discount Amount	Subtotal
[qwerty] Phone Booth	Phone Booth	1.00	1,200.00	IGST	0.0000	₹ 1,200.00

This is how we import an order successfully using our Odoo Shopify Connector.

**Note:** As the Phone booth order has been imported to Odoo, the stock value at the Shopify end will change. Since the stock update is a manual process, you need to update the stock value at the Odoo end manually. Follow the Stock import function as we mentioned in section 4.1 (b). This way your stock associated with a particular product will get updated at the Odoo end.

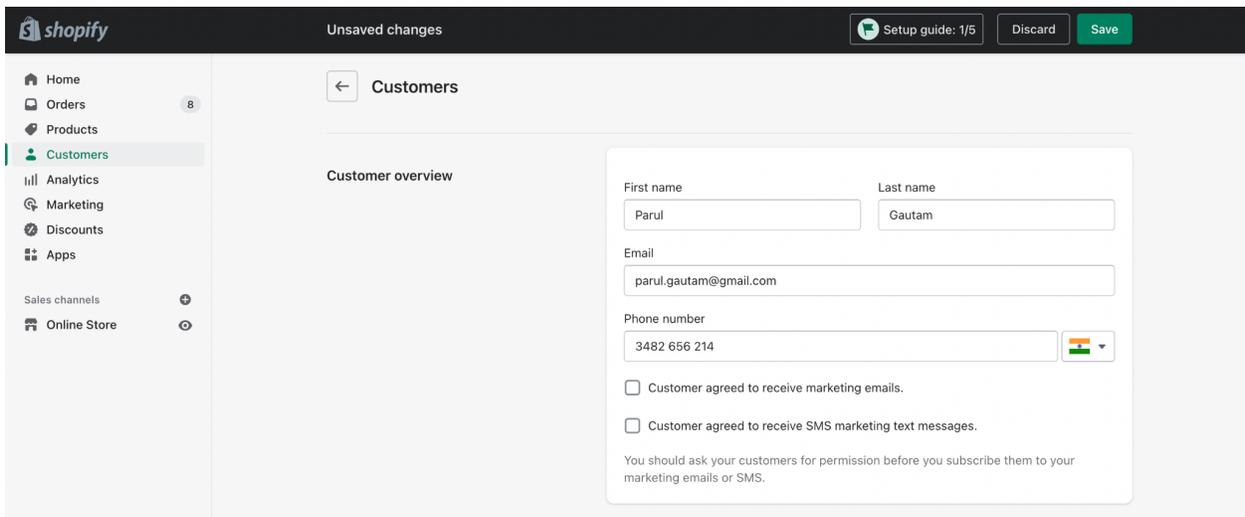
#### d. Import Customer

You can import an already existing customer or create a new one and then export.

Go to Shopify backend>customers>Add Customer



Fill in the Customer Overview details as shown below:



Scroll down and fill in the Address details associated with the customer.

shopify Unsaved changes Setup guide: 1/5 Discard Save

Home Orders Products Customers Analytics Marketing Discounts Apps Sales channels Online Store

**Address**  
The primary address of this customer

Country/region: India

First name: Parul Last name: Gautam

Company: PG Link

Address: New Delhi

Apartment, suite, etc.:

City: New Delhi State: Delhi PIN code: 12348

Phone: 3482 656 214

Give Notes, Tags, Metafields, and click on the Save button at the top right corner.

shopify Unsaved changes Setup guide: 1/5 Discard Save

Home Orders Products Customers Analytics Marketing Discounts Apps Sales channels Online Store

**Tax exemptions**  Collect tax

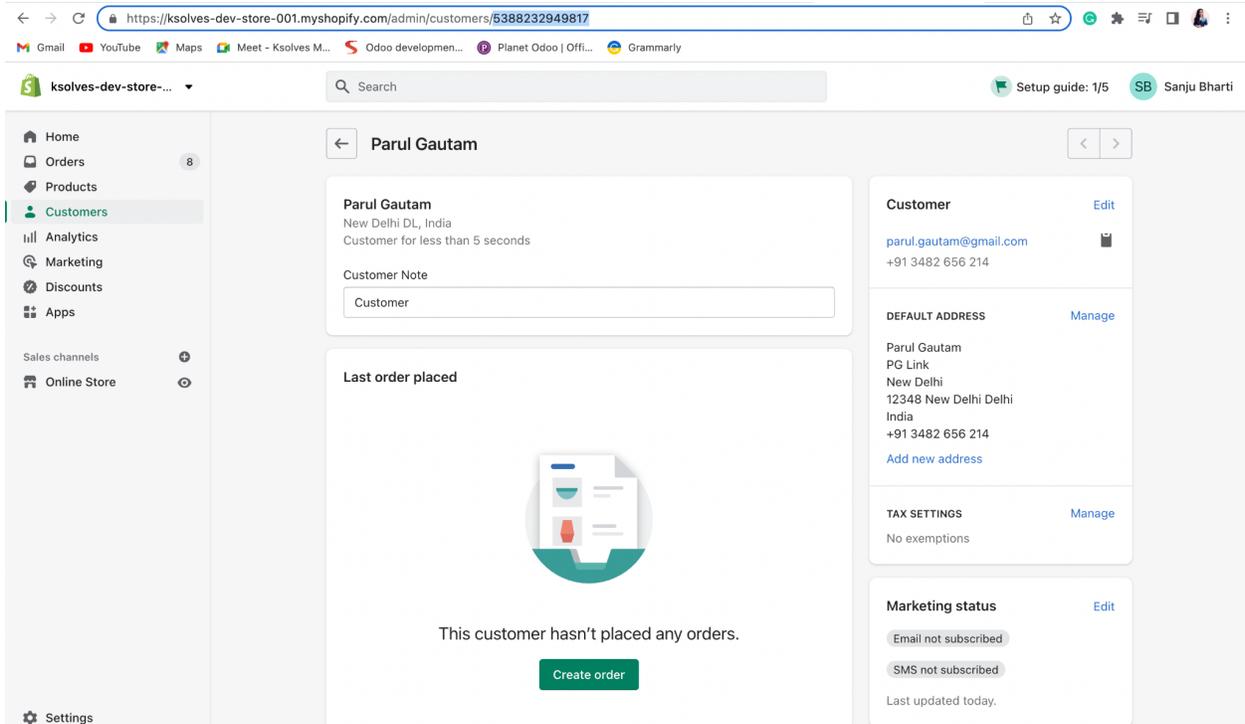
**Notes**  
Add notes about your customer.  
Note: Customer

**Tags**  
Tags can be used to categorize customers into groups.  
The Main Company X  
Add existing tags: fdgd

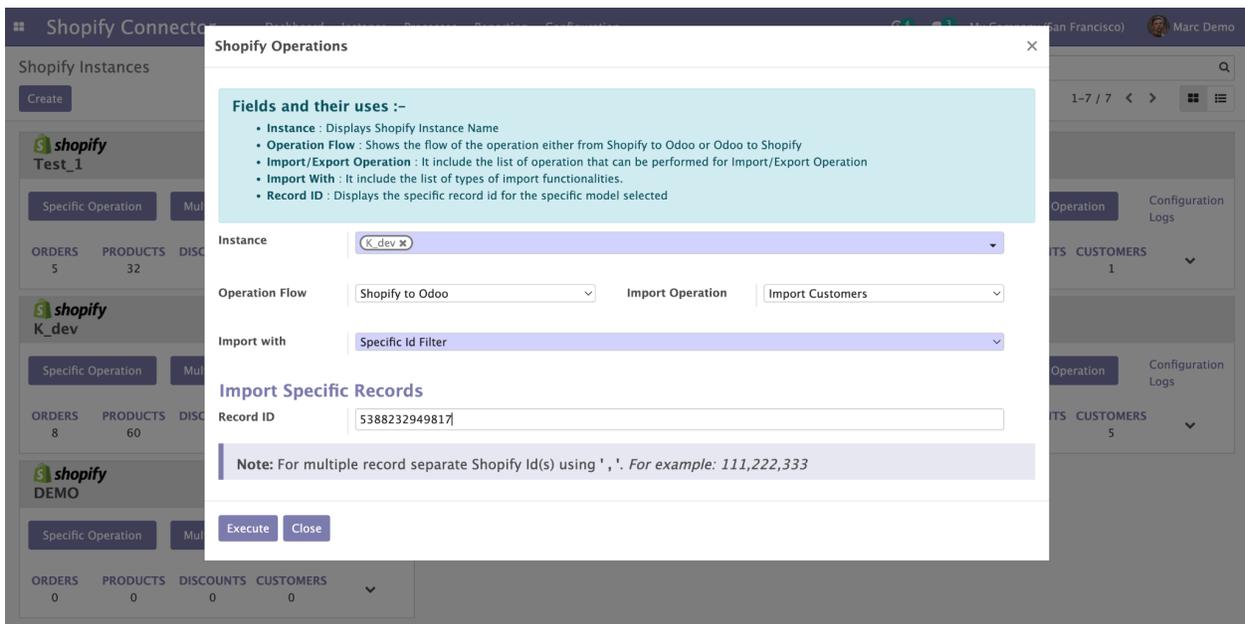
**Metafields**  
These fields have been customized for your store. Apps may need these values to function correctly. [Learn more](#)

tax: Good to Go! Clear  
Single line text  
[Go to definition](#)

Now copy the unique Id of the customer. Here the unique id is 5388232949817.



Go to Odoo Shopify Connector>Instance>Specific Operation  
Use the specific id to import the customer.



On clicking on the Execute button you will get a window as given below:

**Info** ✕

Shopify Operations has been performed, Please refer logs for further details.

[Close](#)

Go to Processes>Queue Jobs. Your customer to be imported will be listed here:

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	New
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Done
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Done
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done

To verify, go to Sales>Orders>Customers

**Sales** My Company (San Francisco) Marc Demo

Customers Customer Invoices Search...

111 111 1111@sss.com \$ 1	30_sept_cus_1_update3 30_sept_cus_1_update3@gmail.com \$ 1	30_sept_cus_1_update3 30_sept_cus_1_update3@gmail.com \$ 1	A.D. Motors Spa Torino, Mongolia noemi.gesu@admotors.it
A.D.Motors Spa Torino, Mongolia noemi.gesu@admotors.it \$ 2	A.D.Motors Spa Torino, Macau noemi.gesu@admotors.it	A.D.Motors Spa, A.D.Motors Spa A.D.Motors Spa Torino, Macau noemi.gesu@admotors.it	Aayush Aarsh Cont@gmail.com \$ 8
Aayush Aarsh aayushaarsh@gmail.com \$ 15	Aayush Aarsh, A a y u s h 1 Aayush Aarsh Noida Cont@gmail.com	Aayush Aarsh, A a y u s h 1 Aayush1 Aayush Aarsh Noida Cont@gmail.com	Aayush Aarsh, Aayush Aarsh Aayush Aarsh NOIDA, India aayushaarsh@gmail.com

Search for your customer and there you will see that your newly created customer has been imported successfully from Shopify to Odoo.

Parul Gautam	
Address	New Delhi New Delhi Delhi (IN) 12348 India
Tax ID	
Job Position	
Phone	+913482656214 SMS
Mobile	
Email	parul.gautam@gmail.com
Website Link	
Title	
Tags	

This is how we can import a customer successfully.

*Similarly, collections, discounts, and other records can be imported easily from Shopify to Odoo. You can consider the DEMO of our product for further import operations.*

## 4.2 Export Operations

### a. Export Product

You can import an existing product or create a new one. Let's export a product by creating a new one in the Odoo.

#### i) Create a Product in Odoo

Creating a New Product in Odoo

Go to Sales>Products>Products>Click on Create button and fill in the required details.

Sales Orders To Invoice Products Reporting Configuration My Company (San Francisco) Marc Demo

Products / New

Save Discard

Product Moves 0.00 Units Sold 0.00 Units Purchased Shopify Products 0.00 Units On Hand 0.00 Units Forecasted 0 Reordering ...

Product Name  
Pen holder

Can be Sold  
 Can be Purchased

General Information Variants Sales Purchase Inventory Images

Product Type: Storable Product  
Product Category: All  
Internal Reference: holder  
Barcode: 78949

Sales Price: \$1.00  
Customer Taxes: Tax 15.00%  
Cost: 0.00  
Company:

Internal Notes  
A new way of writing!

Go to the Inventory tab and add details as required and click on the Save button.

Sales Orders To Invoice Products Reporting Configuration My Company (San Francisco) Marc Demo

Products / [holder] Pen holder

Save Discard 81 / 81

0.00 Units On Hand 0.00 Units Forecasted Product Moves 0 Reordering ... 0.00 Units Sold 0.00 Units Purchased Shopify Products

Product Name  
Pen holder

Can be Sold  
 Can be Purchased

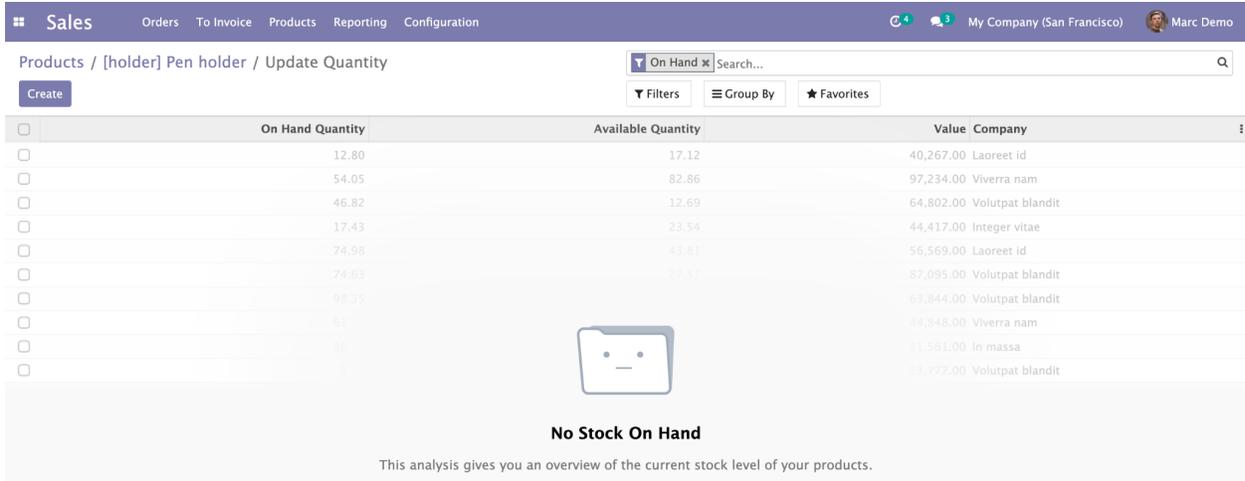
General Information Variants Sales Purchase **Inventory** Images

**Operations**  
Routes: Buy (View Diagram)

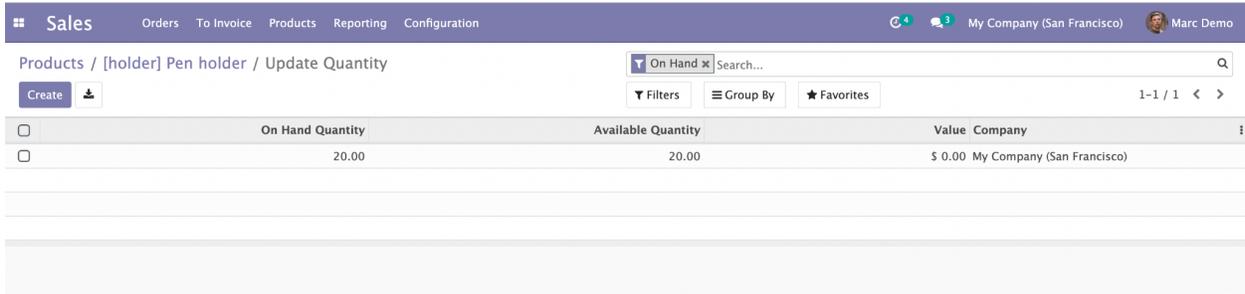
**Logistics**  
Responsible: Marc Demo  
Weight: 12.00 kg  
Length: 0.00 Width: 0.00 Height: 0.00  
Volume: 0.00 m<sup>3</sup>

**Traceability**  
Tracking:  By Unique Serial Number  By Lots  No Tracking

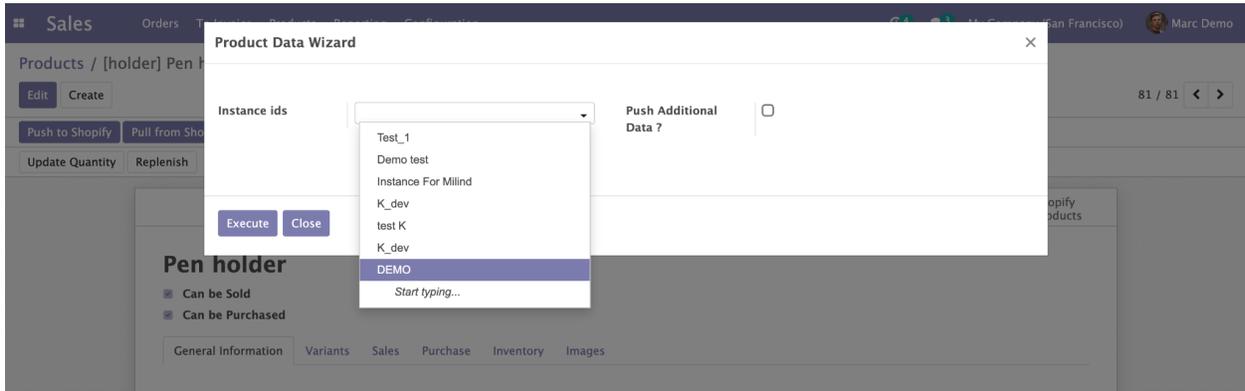
As we can see the On Hand quantity is 0. Let's update the stock value. Click on the On Hand smart button. A window will appear as given below:



Click on Create button and add the quantity as per your choice and click on save. Here we have added 20.

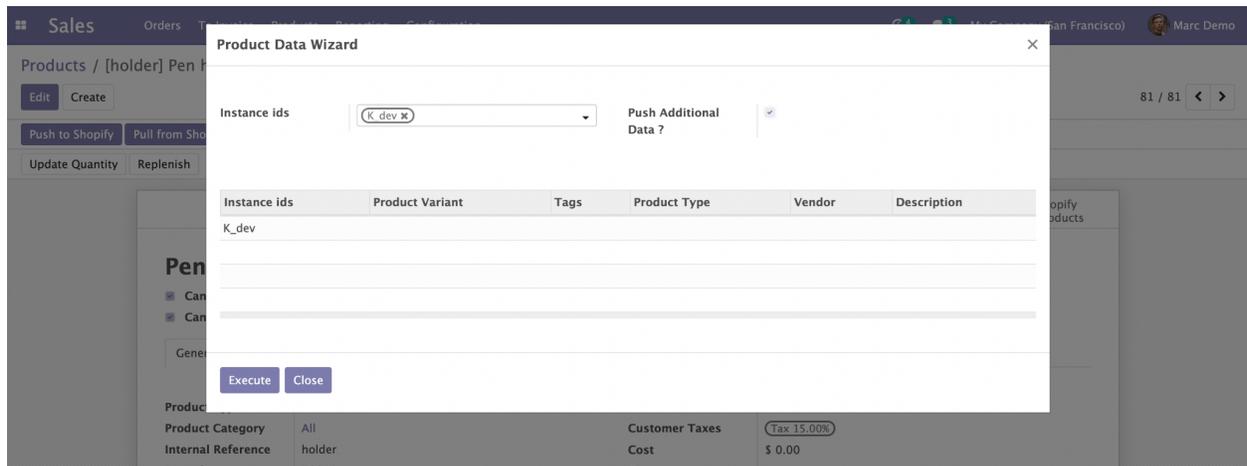


Now go back to your product again and click on the 'Push to Shopify'. As you will click on this button, a wizard will appear as given below:



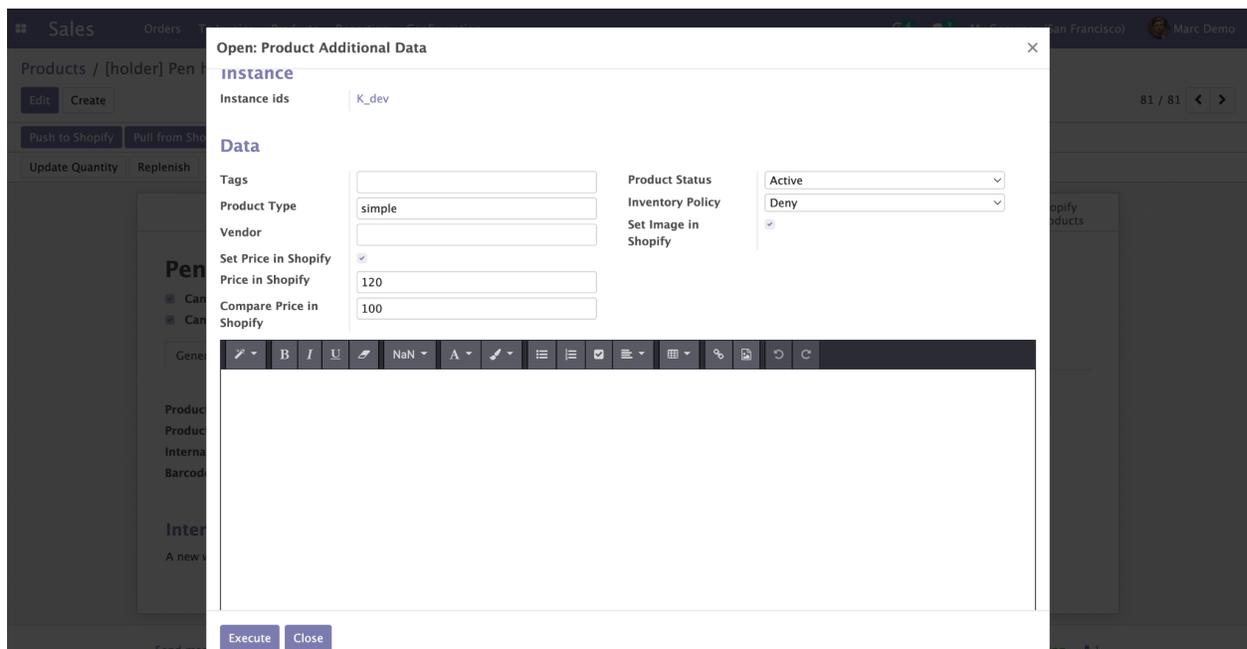
Select the instance id and tick or untick the 'Push Additional Data' as per the choice.

Once you fill in the details, you can see a layer model of your instance id is created.



On clicking on the instance id layer model, the product additional data window will open. You can select the Tags, product type, set image in Shopify, set price, vendor details, product status, and description.

Once done, click on Execute>Execute



Now your product to be exported has been enqueued in the Queue Job. Go to Processes>Queue Jobs. here you will find the record.

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
[holder] Pen holder	6731024924729	Product Template	Shopify Layer to Shopify	Export	K_dev	Done
Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
Sale Order #4014983643193	4014983643193	Sale Order	Shopify To Odoo	Import	K_dev	Done
Sale Order #4014983643193	4014983643193	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Sale Order #4014983643193	4014983643193	Sale Order	Shopify To Odoo	Import	K_dev	Done
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
Sale Order #4014983643193	4014983643193	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
Phone Booth	6724604854329	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Done
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Done
Sale Order #4014979022905	4014979022905	Sale Order	Shopify To Odoo	Import	K_dev	Failed

To verify, visit the Shopify backend, go to Products, and search for the product name.

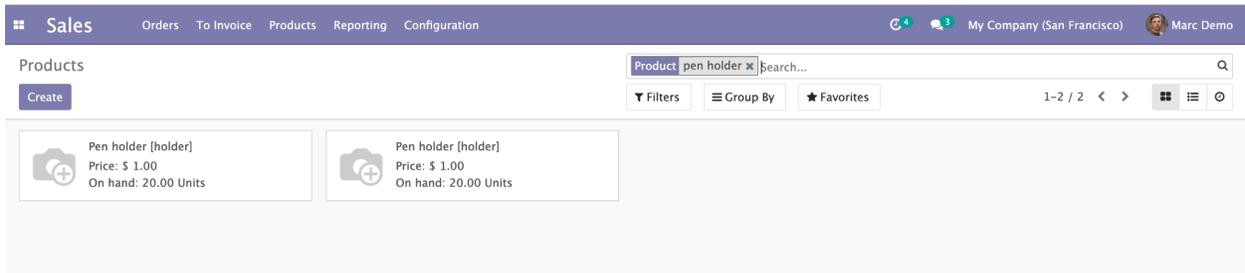
For example, we have exported a ‘pen holder’, and here we can see that our product is successfully exported.

Product	Status	Inventory	Type	Vendor
Pen holder	Active	20 in stock	simple	ksolves-dev-store-001
Pen holder	Active	0 in stock	simple	ksolves-dev-store-001

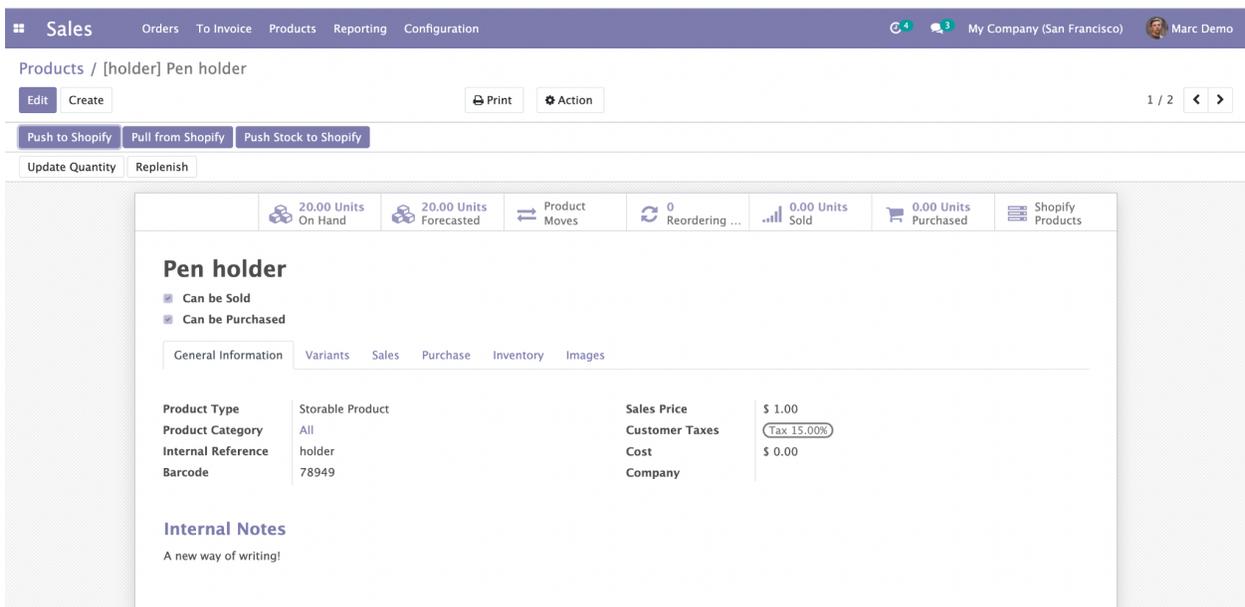
[Add a screenshot of the data such as Shopify ID, created and synced date (in the layer) being updated after the successful export]

## b. Export Product Stock

Go to Sales>Products>Products and search for a product whose stock you need to export.



Open your product and update its stock value.



For example, here we have updated the stock value from 20 to 30.

Sales Orders To Invoice Products Reporting Configuration My Company (San Francisco) Marc Demo

Products / [holder] Pen holder / Update Quantity

On Hand Search... Filters Group By Favorites 1-1 / 1 < >

	On Hand Quantity	Available Quantity	Value	Company
<input type="checkbox"/>	30.00	30.00	\$ 0.00	My Company (San Francisco)

Now, come back to the product and click on the 'Push Stock to Shopify'.

Sales Orders To Invoice Products Reporting Configuration My Company (San Francisco) Marc Demo

Products / [holder] Pen holder

Edit Create Print Action 1 / 2 < >

Push to Shopify Pull from Shopify Push Stock to Shopify

Update Quantity Replenish

30.00 Units On Hand 30.00 Units Forecasted Product Moves 0 Reordering ... 0.00 Units Sold 0.00 Units Purchased Shopify Products

### Pen holder

Can be Sold  
 Can be Purchased

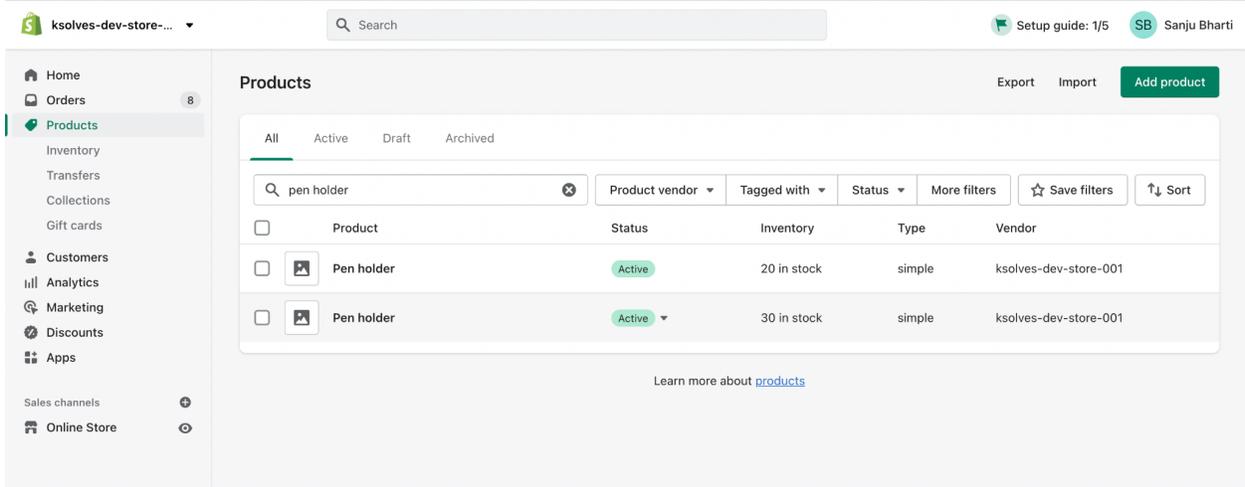
General Information Variants Sales Purchase Inventory Images

Product Type	Storable Product	Sales Price	\$ 1.00
Product Category	All	Customer Taxes	Tax 15.00%
Internal Reference	holder	Cost	\$ 0.00
Barcode	78949	Company	

**Internal Notes**  
A new way of writing!

This way your updated stock value will be exported at the Shopify end as well.

To verify, go to the Shopify end and check the product stock.

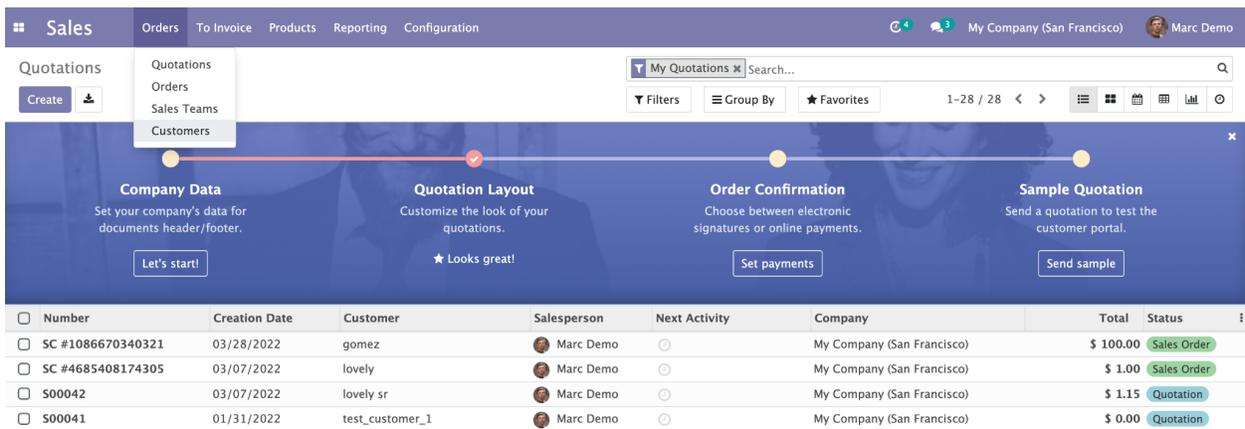


You can export stock of every storable product in the same way as explained in this section.

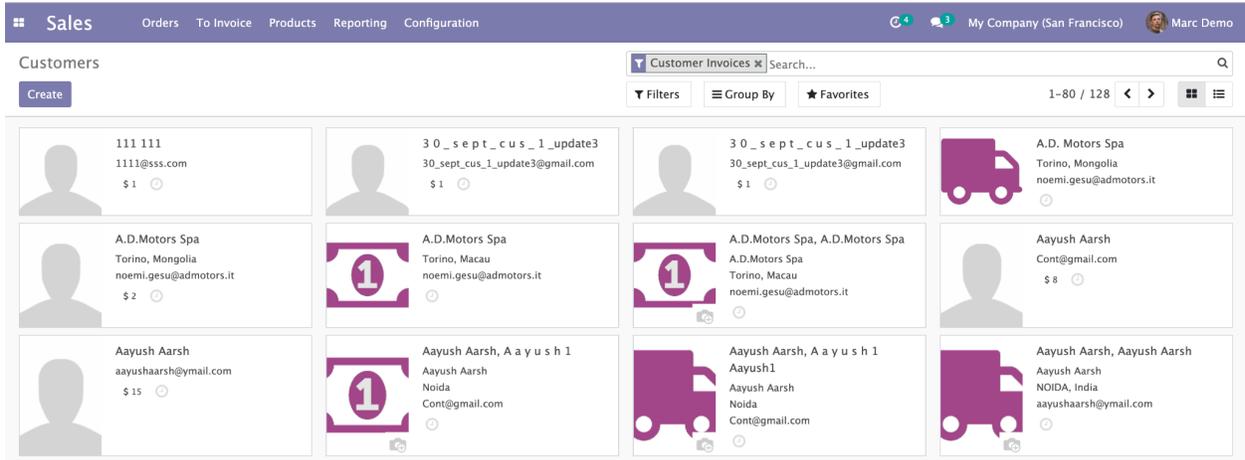
### c. Export Customer

You can import an already existing customer or create a new one and then export.

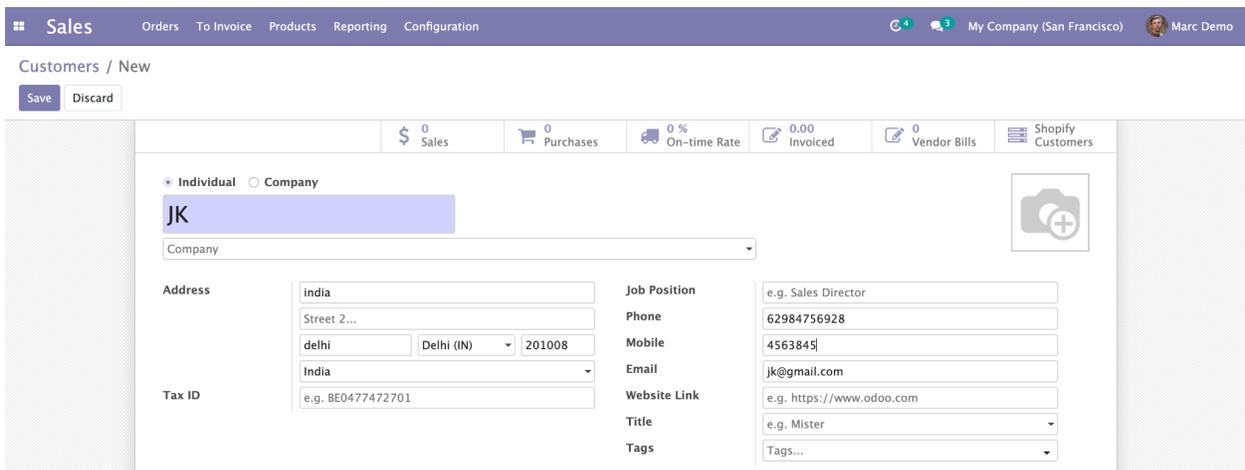
Go to Sales>Orders>Customers



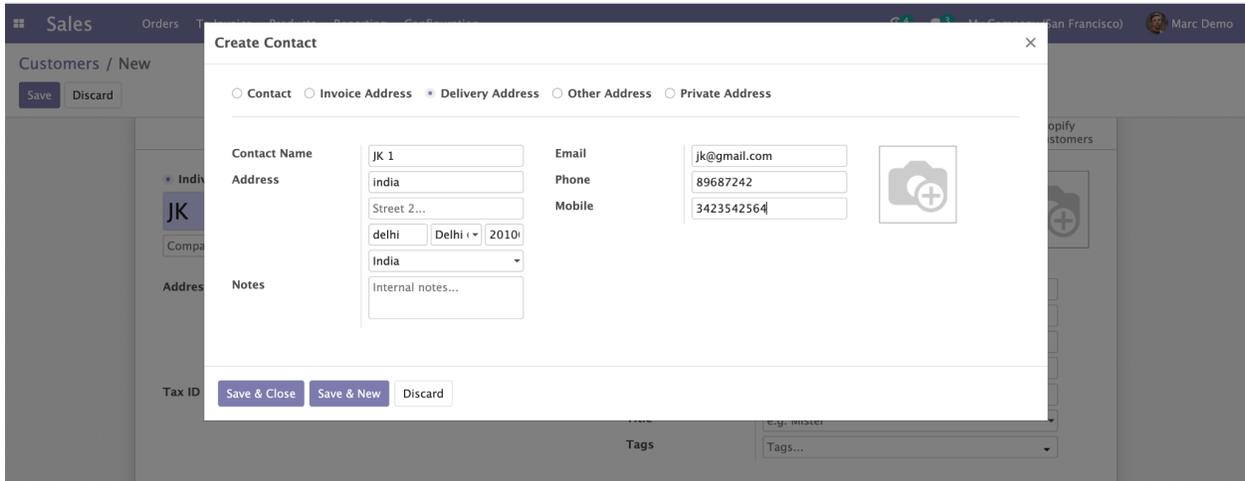
Click on the create button.



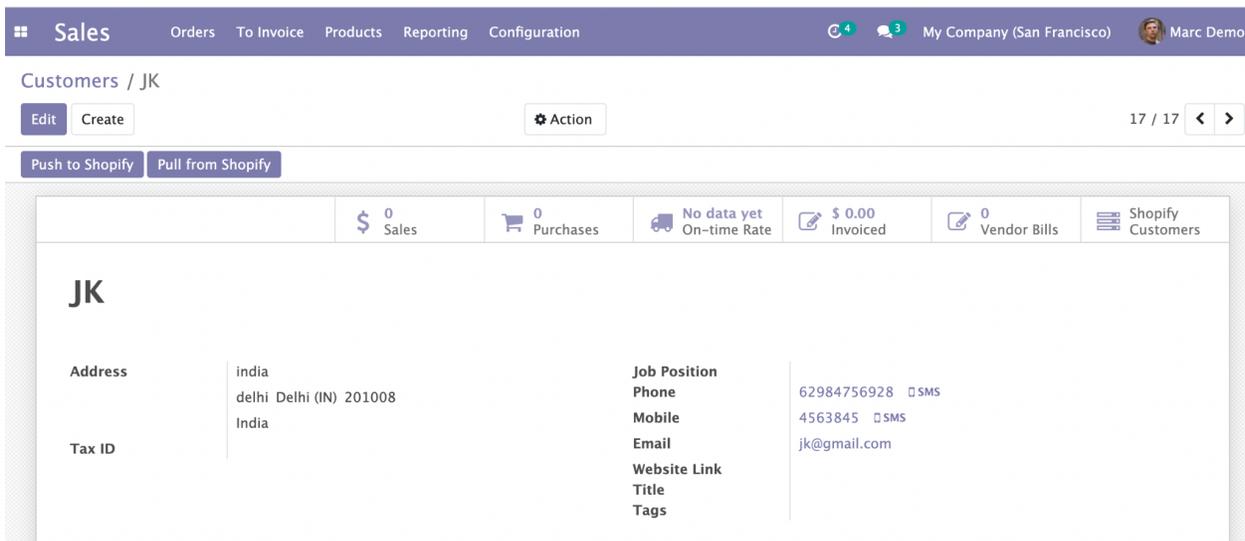
Fill in the required fields related to a new customer. Click on the Add button presented at the bottom.



A create contact window will appear. Fill in the necessary fields. Click on the Save & Close button. Now click on the Save button.

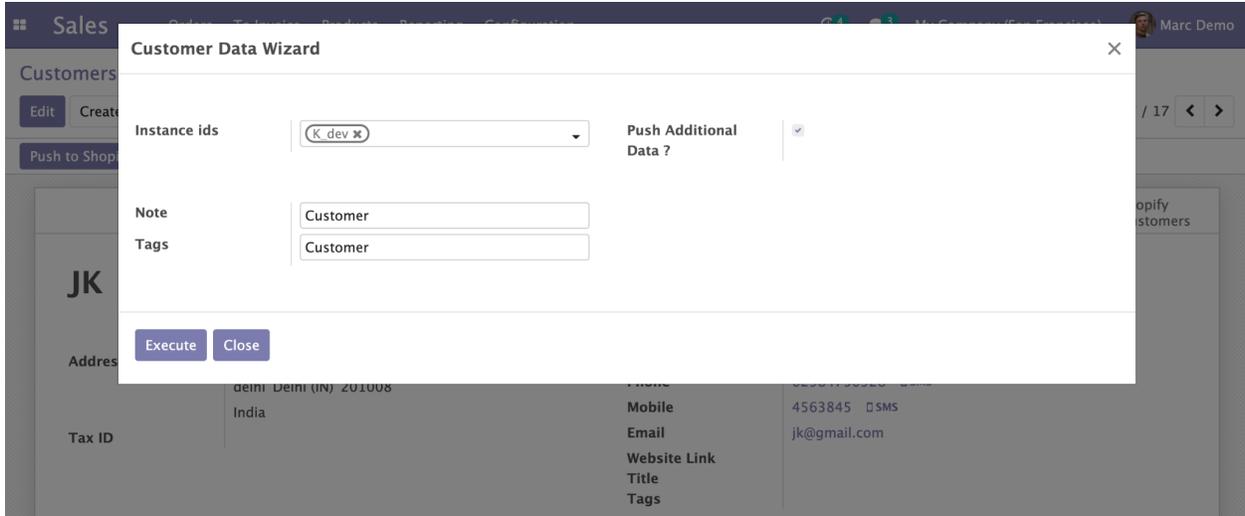


Click on the 'Push to Shopify' button.



A wizard will open. Mention the Instance Id and other details.

Then click on the Execute button.



**Note:-** A single customer when exported, no record is enqueued in the queue. Rather when customers(>1) are exported together, in that case, all the customers to be exported are enqueued in the Queue Jobs.

Now to check the exported customer status, go to **Processes>Success Logs**.

There you can find your newly exported customer with the date and time details as well.

Log Id	Domain	Operation Performed	Operation Status	Operation Flow	Created on	Shopify Instance
LOG184341	Customer	Prepare Update	Success	Odoo to Shopify	04/18/2022 19:20:56	K_dev
LOG184338	API Data Handlings	Create	Success	Shopify Layer to Shopify	04/18/2022 19:20:56	K_dev
LOG184339	Customer	Create	Success	Odoo to Shopify	04/18/2022 19:20:56	K_dev
LOG184340	Customer	Create	Success	Odoo to Shopify	04/18/2022 19:20:56	K_dev
LOG184337	Customer	Prepare Update	Success	Odoo to Shopify	04/18/2022 19:17:57	K_dev
LOG184335	Customer	Prepare Create	Success	Odoo to Shopify	04/18/2022 19:14:05	K_dev
LOG184333	Customer	Prepare Create	Success	Odoo to Shopify	04/18/2022 19:13:15	K_dev

If you click on the customer, you will get a banner showing the 'SUCCESS' of the export customer.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo

Success Logs / LOG184341

Action 1 / 80

**Fields and their uses :-**

- **Operation Performed** : Displays operation type which is performed
- **Created on** : Displays the created date
- **Operation Status** : Displays the status of the operation Success/Failed
- **Operation Flow** : Shows the flow of the operation either from Shopify to Odoo or Odoo to Shopify
- **Domain** : Shows name of the model
- **Shopify Instance** : Displays Shopify Instance Name
- **Layer Model Associated** : Displays the layer model which is associated
- **Odoo Model Associated** : Displays the odoo default model which is associated
- **Odoo Record ID** : Displays the odoo record ID
- **Summary** : Displays the Summary of the Logs

**LOG184341**

<b>Basic Info</b>		<b>Status</b>	
Operation Performed	Prepare Update	Operation Status	Success
Created on	04/18/2022 19:20:56	Operation Flow	Odoo to Shopify
<b>Shopify Information</b>		<b>Model Info</b>	
Domain	Customer	Odoo Model Associated	Contact
Shopify Instance	K dev		

To verify, go to Shopify backend>Customer>Search for your customer.

ksolves-dev-store-... Search Setup guide: 1/5 SB Sanju Bharti

All customers ▾ Export Import Email segment Add customer

16 customers No filters applied

To create a segment, apply a filter or choose a template. Find a customer using the top search bar.

Showing 16 of 16 customers Sort

<input type="checkbox"/>	Customer name	Status	Location	Orders	Spent
<input type="checkbox"/>	JK		delhi DL, India	0 orders	₹0.00
<input type="checkbox"/>	Parul Gautam		New Delhi DL, India	2 orders	₹1,417.15
<input type="checkbox"/>	Prateek shah		noida UP, India	0 orders	₹0.00

This is how you can easily export a customer from the Odoo end.

## d. Export Order

Go to Sales> click on Create button.

Sales | Orders | To Invoice | Products | Reporting | Configuration | My Company (San Francisco) | Marc Demo

Quotations

My Quotations | Search...

Filters | Group By | Favorites | 1-28 / 28

**Company Data**  
Set your company's data for documents header/footer.

[Let's start!](#)

**Quotation Layout**  
Customize the look of your quotations.

★ Looks great!

**Order Confirmation**  
Choose between electronic signatures or online payments.

[Set payments](#)

**Sample Quotation**  
Send a quotation to test the customer portal.

[Send sample](#)

Number	Creation Date	Customer	Salesperson	Next Activity	Company	Total	Status
SC #1086670340321	03/28/2022	gomez	Marc Demo		My Company (San Francisco)	\$ 100.00	Sales Order
SC #4685408174305	03/07/2022	lovely	Marc Demo		My Company (San Francisco)	\$ 1.00	Sales Order
S00042	03/07/2022	lovely sr	Marc Demo		My Company (San Francisco)	\$ 1.15	Quotation
S00041	01/31/2022	test_customer_1	Marc Demo		My Company (San Francisco)	\$ 0.00	Quotation
S00038	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 116.15	Quotation
S00036	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 1.15	Cancelled
S00031	10/06/2021	Farooq Nasim	Marc Demo		My Company (San Francisco)	\$ 115,000.00	Quotation
S00028	10/06/2021	Gaurav Kumar	Marc Demo		My Company (San Francisco)	\$ 17,250.00	Sales Order

A window will appear that will ask you to fill in the required details. Fill in the customer name, and address, and add the product in the 'Order Line' tab.

Sales | Orders | To Invoice | Products | Reporting | Configuration | My Company (San Francisco) | Marc Demo

Quotations / New

Save | Discard

Customer Preview

UnSynchron... With Shopify

### New

Customer: Parul Gautam | Expiration: [ ]

Invoice Address: Parul Gautam | Payment Terms: [ ]

Delivery Address: Parul Gautam | Auto Workflow Process: [ ]

Quotation Template: [ ]

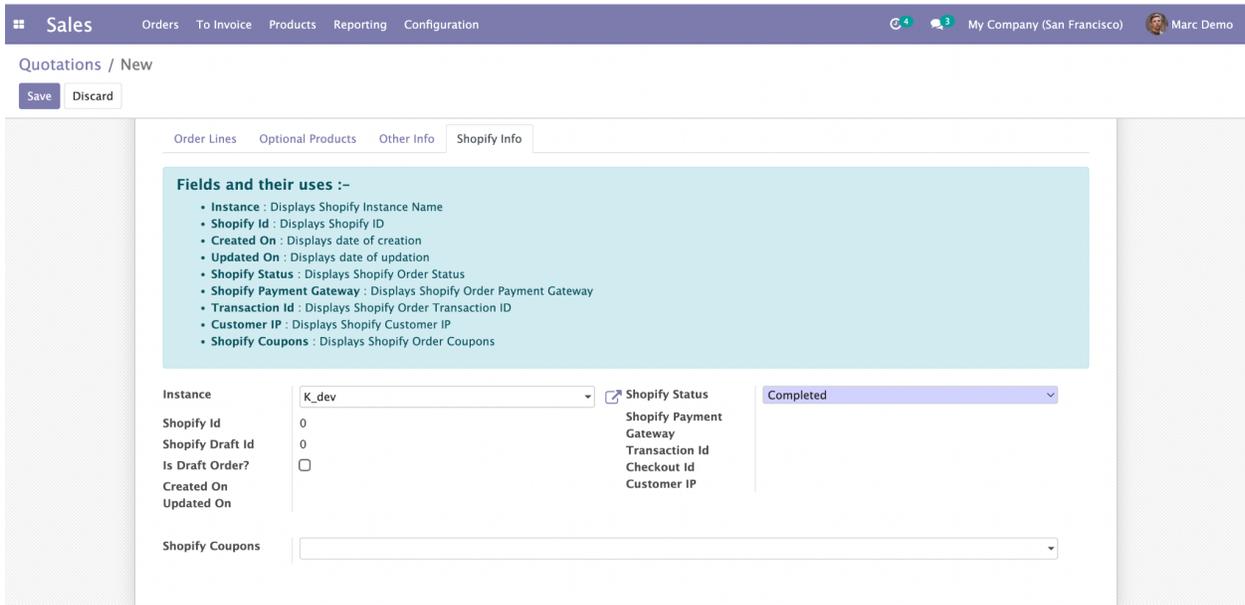
Order Lines | Optional Products | Other Info | Shopify Info

Product	Description	Quantity	Unit Price	Taxes	Discount ...	Subtotal
[qwerty] Phone Booth	[qwerty] Phone Booth	1.00	1.00	Tax 15.00%	0.0000	\$ 1.00

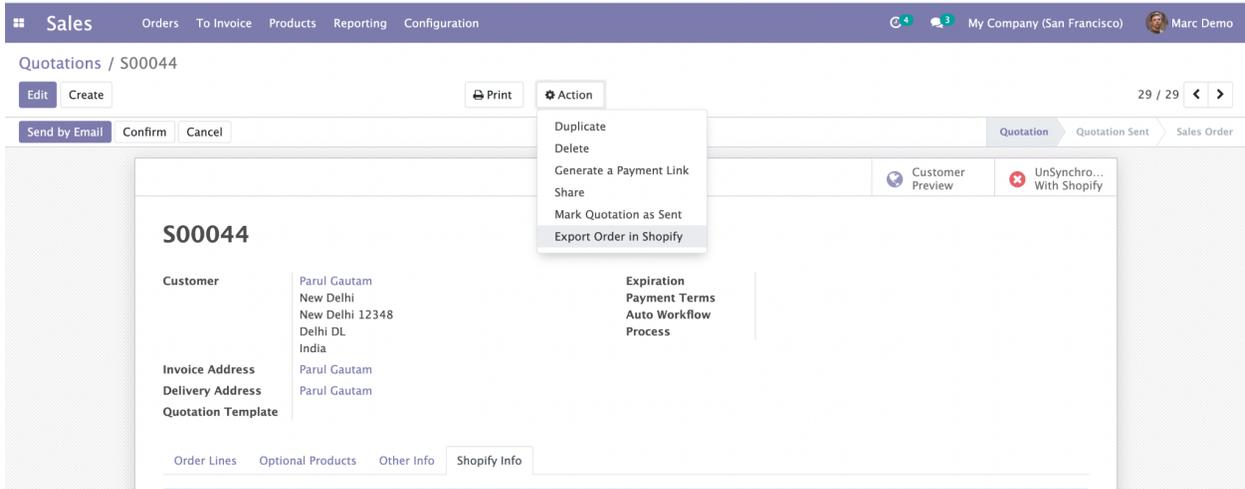
Terms and conditions...

Untaxed Amount: \$ 1.00  
Taxes: \$ 0.15  
Total: \$ 1.15

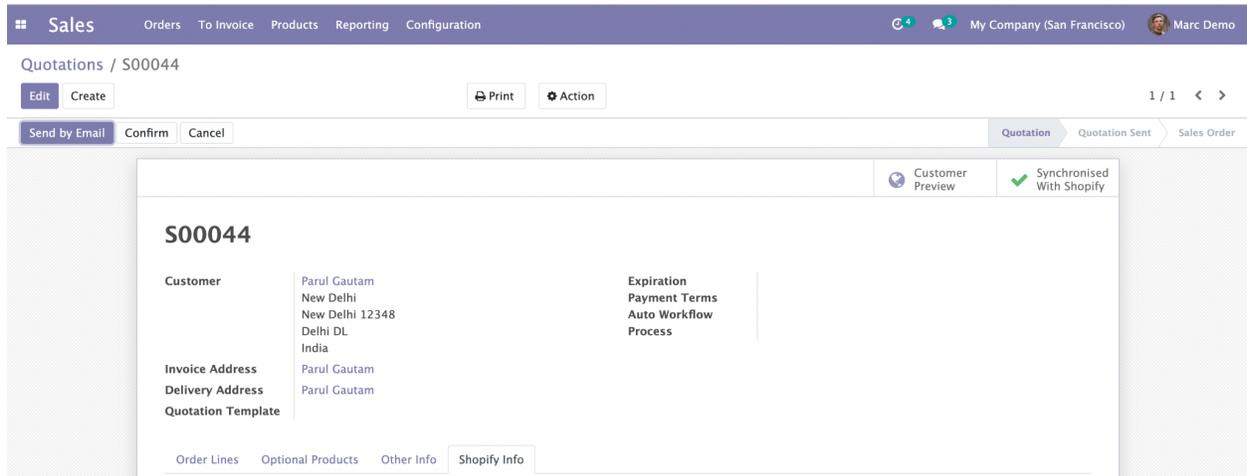
After filling in the details on the first page, go to the 'Shopify Info' tab. Select the Instance and Shopify Status, then click on the 'Save' button.



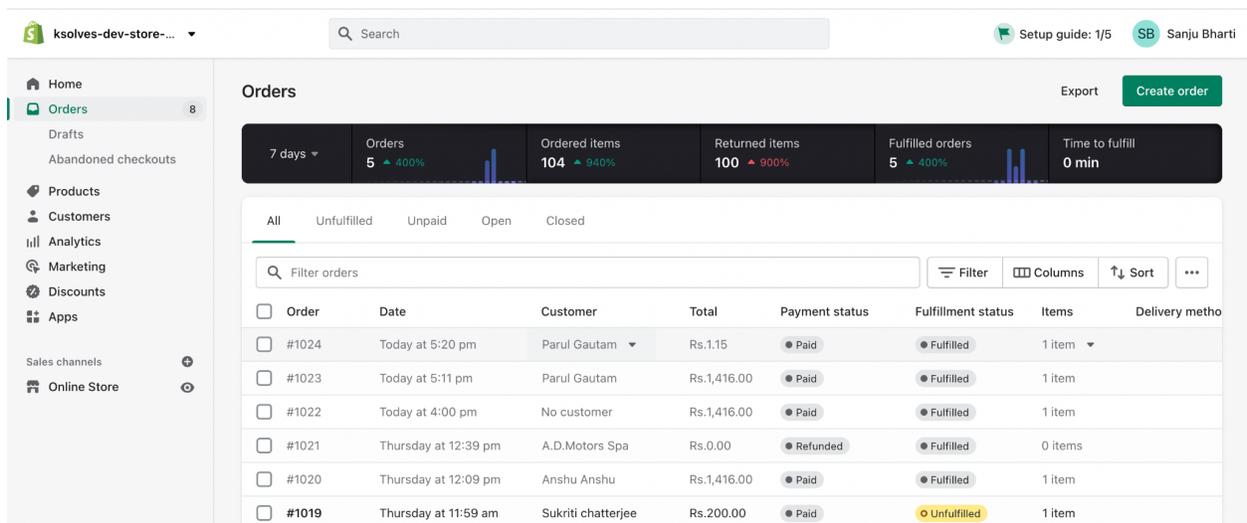
Click on the Action button and then click on the 'Export Order to Shopify'. As you can see at the top right corner, the order is in 'Unsynchronized with Shopify'.



As soon as you export it, the (X) mark will change to the right tick, reflecting that the order has been successfully exported to Shopify.



To verify, go to the Shopify backend, click on Orders, and there you will find your recently exported order in the list.



This is how the Order export function is performed.

#### 4.2.1 Export Records in Bulk

Records can be exported in bulk using the Instance from Odoo to Shopify.

**Note:** Before exporting records in bulk, every record must be synced once between Shopify and Odoo.

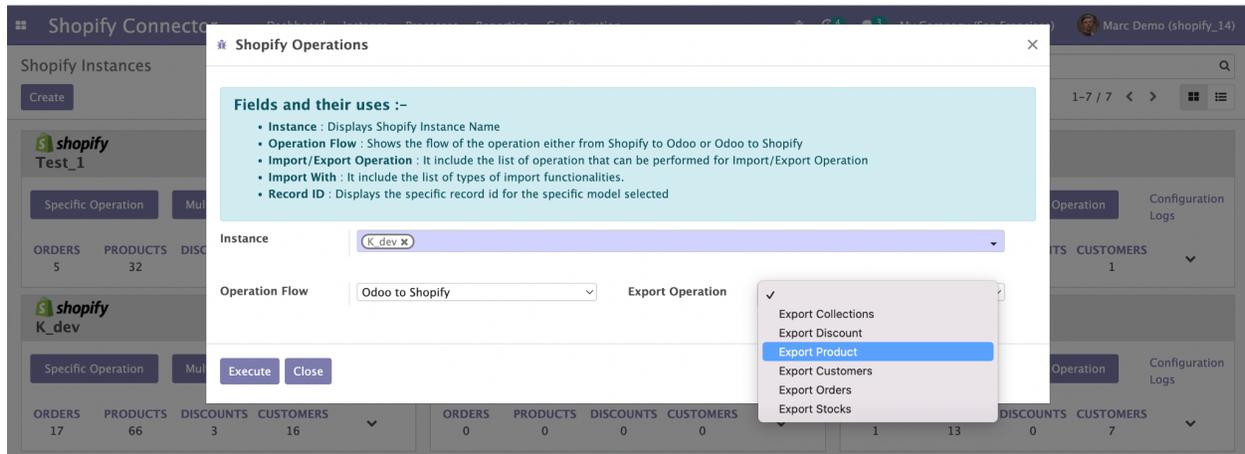
Let's see how to perform an export function in bulk.

Go to Instance>Specific Operation

Operation Flow: Odoo to Shopify

Export Operation: Select the record which you want to export.

Click on the Execute button.



To check the exported records, go to Processes>Queue jobs

The screenshot shows the 'All Queue Jobs' table in the Shopify Connector application. The table has the following columns: Name, Shopify ID, Domain, Operation Flow, Operation Type, Shopify Instance, and State. The table contains 20 rows of data, all with a 'New' state.

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
[qwerty] Phone Booth	6724604854329	Product Template	Shopify Layer to Shopify	Export	K_dev	New
[tee] Tshirt White	6724627136569	Product Template	Shopify Layer to Shopify	Export	K_dev	New
[holder] Pen holder	6724630118457	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Tshirts	6724630642745	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Tshirts	6724630577209	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Tshirts	6724630544441	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Curios Sweatshirt in Steel Grey	6724630839353	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Jeth Sweatshirt in Blue/Royal	6724630806585	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Loopwheeler Sweatshirt in Black	6724630741049	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Jeth Sweatshirt in Grey/Rust	6724630773817	Product Template	Shopify Layer to Shopify	Export	K_dev	New
tshirt001	6724630478905	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Symphonie Sweatshirt in Black	6724630675513	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Loopwheeler Sweatshirt in Grey	6724630708281	Product Template	Shopify Layer to Shopify	Export	K_dev	New
tshirt002	6724630446137	Product Template	Shopify Layer to Shopify	Export	K_dev	New
tshirt003	6724630413369	Product Template	Shopify Layer to Shopify	Export	K_dev	New
Vintage Sweatshirt in Light Grey	6724630511673	Product Template	Shopify Layer to Shopify	Export	K_dev	New
[klij] Stands	6726885572665	Product Template	Shopify Layer to Shopify	Export	K_dev	New

Thus, exporting records in bulk is extremely easy with our Odoo Shopify Connector

If you go to Sales, select all the Orders, click on the Action button and choose the Export Order In Shopify, this way also we can export orders in bulk.

Number	Creation Date	Customer	Salesperson	Next Activity	Company	Total	Status
SC #1086670340321	03/28/2022	gomez	Marc Demo		My Company (San Francisco)	\$ 100.00	Sales Order
SC #4685408174305	03/07/2022	lovely	Marc Demo		My Company (San Francisco)	\$ 1.00	Sales Order
S00042	03/07/2022	lovely sr	Marc Demo		My Company (San Francisco)	\$ 1.15	Quotation
S00041	01/31/2022	test_customer_1	Marc Demo		My Company (San Francisco)	\$ 0.00	Quotation
S00038	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 116.15	Quotation
S00036	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 1.15	Cancelled
S00031	10/06/2021	Farooq Nasim	Marc Demo		My Company (San Francisco)	\$ 115,000.00	Quotation
S00028	10/06/2021	Gaurav Kumar	Marc Demo		My Company (San Francisco)	\$ 17,250.00	Sales Order
S00027	10/03/2021	Gemini Furniture	Marc Demo		My Company (San Francisco)	\$ 14.00	Sales Order
S00026	10/03/2021	trtx_one, delivery new	Marc Demo		My Company (San Francisco)	\$ 400.00	Sales Order
S00024	09/17/2021	o d o o _ c u s _ n e w _ a	Marc Demo		My Company (San Francisco)	\$ 11.50	Quotation
S00023	08/27/2021	Azure Interior	Marc Demo		My Company (San Francisco)	\$ 320.00	Sales Order
S00022	08/13/2021	Gemini Furniture	Marc Demo		My Company (San Francisco)	140.00 €	Sales Order

### 4.3 Auto Configure Schedulers

We call them either configure schedulers or cron jobs in Odoo.

Go to Instance>Configuration.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Shopify Instances / K\_dev

Deactivate Instance

Price lists Configure Schedulers Active Schedulers

**Fields and their uses :-**

- Connector Instance Name : Displays Shopify Instance Name
- Company : Shows the name of the company

**K\_dev**

Company My Company (San Francisco)

Require Meta Mapping ?

Authentication Currency Information Product Information Order Information Tax Information Webhooks Configuration Meta Mappings

**Fields and their uses :-**

- Store URL : Displays the Shopify Store URL
- Customer Key : Customer Key of the Shopify, not visible by default
- Customer Password : Customer Password of the Shopify, not visible by default
- Shared Secret : Shared Secret of the Shopify, not visible by default
- Shop ID : Shop ID of the Shopify Shop, not visible by default

ACTIVATED

These are all the configured schedulers:

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Shopify Instances / K\_dev / Shopify Schedulers

Search...

Filters Group By Favorites 1-6 / 6

Priority	Action Name	Model	Next Execution Date	Interval Number	Interval Unit	Number of Calls	Active
5	KS: Shopify Sales Report	Shopify Mail Report Configuration	04/19/2022 21:21:29	24	Hours	-1	<input checked="" type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Customer Import from Shopify to ...	Custom Partner to Connect Multiple Connectors	04/20/2022 10:23:07	1	Days	-1	<input type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Order Import from Shopify to Od...	Sales Order	04/20/2022 10:23:07	1	Days	-1	<input checked="" type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Product Export from Odoo to Sh...	Shopify Product Model	04/20/2022 10:23:07	1	Days	-1	<input type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Product import from Shopify to O...	Shopify Product Model	04/13/2022 10:23:07	1	Days	-1	<input checked="" type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Product Sync from Shopify to Odo...	Shopify Auto Product Syncing Configuration	04/20/2022 10:23:07	1	Days	-1	<input type="checkbox"/>

And the active one can be visible from the Active Schedulers from the instance.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Shopify Instances / K\_dev

Deactivate Instance

Price lists Configure Schedulers Active Schedulers

**Fields and their uses :-**

- Connector Instance Name : Displays Shopify Instance Name
- Company : Shows the name of the company

**K\_dev**

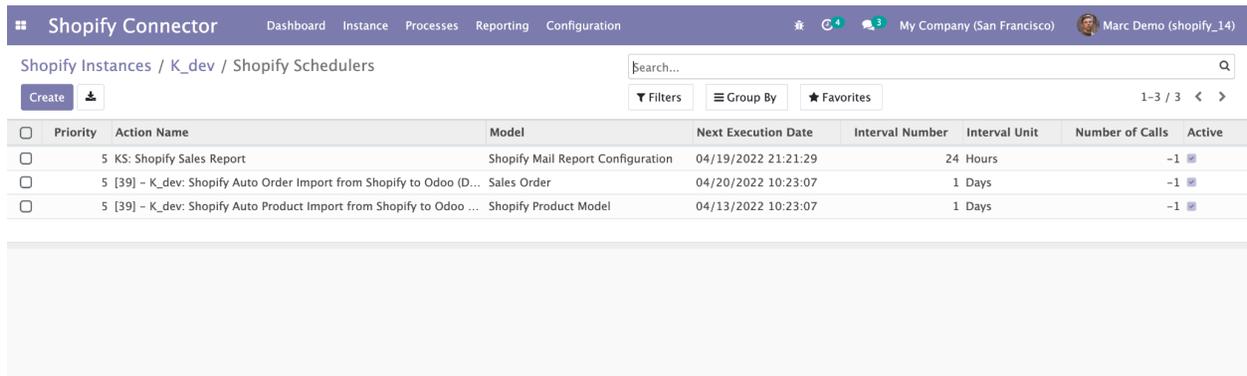
Company My Company (San Francisco)

Require Meta Mapping ?

Authentication Currency Information Product Information Order Information Tax Information Webhooks Configuration Meta Mappings

ACTIVATED

These are all the Active schedulers:

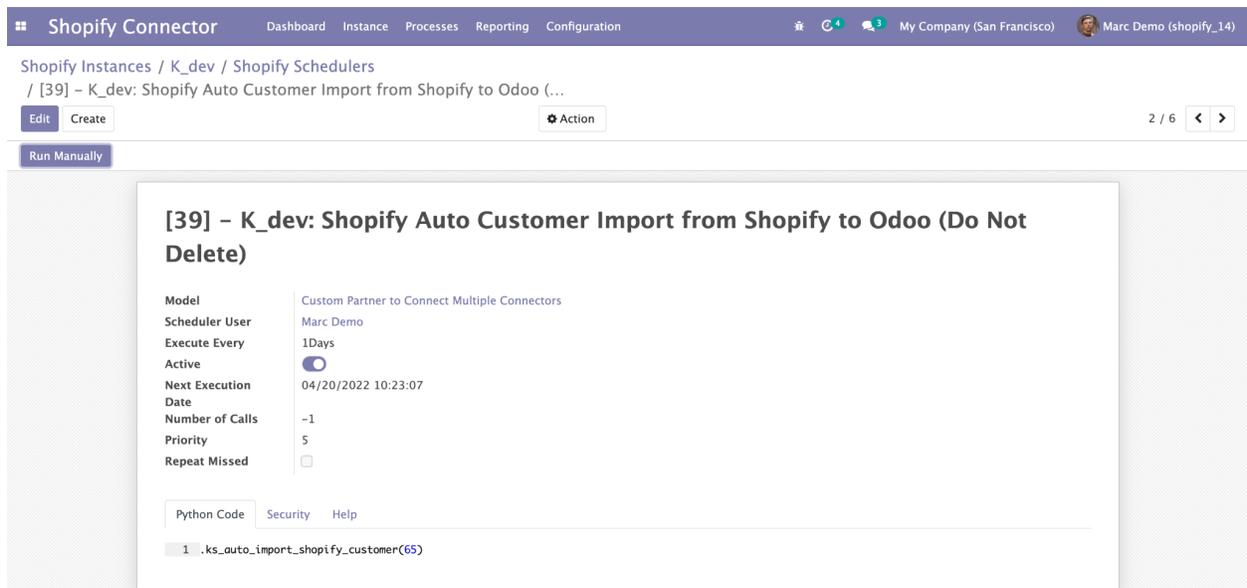


The screenshot shows the 'Shopify Schedulers' page in the Shopify Connector application. It features a table with columns for Priority, Action Name, Model, Next Execution Date, Interval Number, Interval Unit, Number of Calls, and Active status. There are three entries listed, all with a priority of 5 and an active status of -1.

Priority	Action Name	Model	Next Execution Date	Interval Number	Interval Unit	Number of Calls	Active
5	KS: Shopify Sales Report	Shopify Mail Report Configuration	04/19/2022 21:21:29	24	Hours	-1	<input checked="" type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Order Import from Shopify to Odoo (D...	Sales Order	04/20/2022 10:23:07	1	Days	-1	<input checked="" type="checkbox"/>
5 [39]	- K_dev: Shopify Auto Product Import from Shopify to Odoo ...	Shopify Product Model	04/13/2022 10:23:07	1	Days	-1	<input checked="" type="checkbox"/>

So, these are all the cron jobs that are created for Shopify. Cron jobs are scheduled actions that run after a specific time and will perform the designated actions.

For example, if we open one configure scheduler, we will get all the details associated with it.



The screenshot shows the configuration details for the scheduler '[39] - K\_dev: Shopify Auto Customer Import from Shopify to Odoo (Do Not Delete)'. The configuration includes the model name, scheduler user, execution frequency, active status, next execution date, number of calls, priority, and repeat missed status. Below the configuration, there is a section for the Python code used for the scheduler.

**[39] - K\_dev: Shopify Auto Customer Import from Shopify to Odoo (Do Not Delete)**

**Model:** Custom Partner to Connect Multiple Connectors

**Scheduler User:** Marc Demo

**Execute Every:** 1Days

**Active:**

**Next Execution Date:** 04/20/2022 10:23:07

**Number of Calls:** -1

**Priority:** 5

**Repeat Missed:**

**Python Code:** `1 .ks_auto_import_shopify_customer(65)`

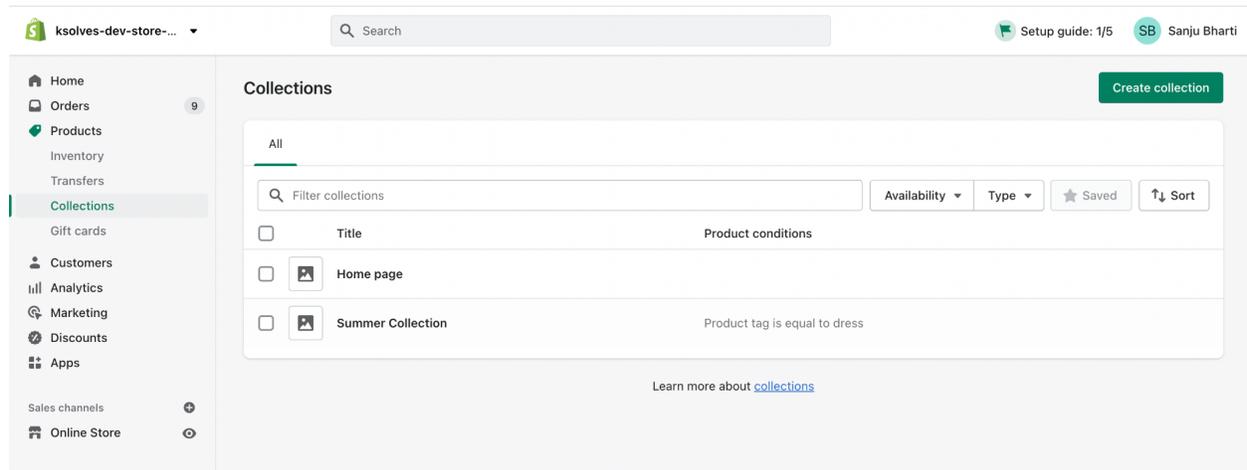
Here, in the above image, the scheduler represents that after every 1 day, Shopify customers will be automatically imported to Odoo.

This is helpful to reduce the time for manually performing import operations.

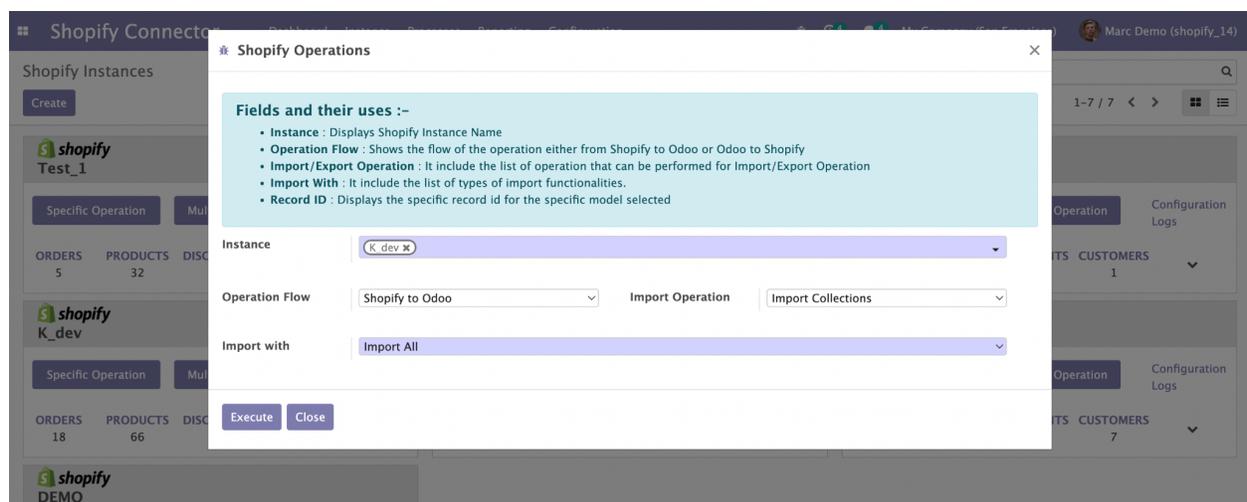
## 4.4 Product/Order Associated Operations

### 4.4.1 Collections

Go to Shopify>Products>Collections. You can choose a collection you wish to import or you can create your new one by clicking on the ‘Create Collection’ button.



Once you create a collection, go to Shopify Instance>Specific Operation>Import Collections and click on the Execute button.



To verify, go to Processes>Queue Jobs. here you can find your imported collection from Shopify to Odoo.

<input type="checkbox"/>	Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/>	Home page	268764741689	Collections	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	dif phase 3	63720161337	Location	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/>	Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/>	Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Done

Go to Dashboard>Collections>Open your collection>and click on the Edit button.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Custom Collections / Home page

Save Discard 1 / 27

### Connector Basic Information

Shopify Collection ID: 268764741689

Collection Name: Home page

Shopify Handle: frontpage

### Date Logs

Date Created: 04/20/2022 18:30:18

Date Updated: 04/20/2022 18:30:18

Body

### Instance Information

Shopify Instance: K\_dev

Company: My Company (San Francisco)

Product Ids:

Add products in the Product Ids section and Save.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Custom Collections / Home page

Edit Create Action 1 / 27

### Connector Basic Information

Shopify Collection ID: 268764741689

Collection Name: Home page

Shopify Handle: frontpage

### Date Logs

Date Created: 04/20/2022 18:30:18

Date Updated: 04/20/2022 18:30:18

Body

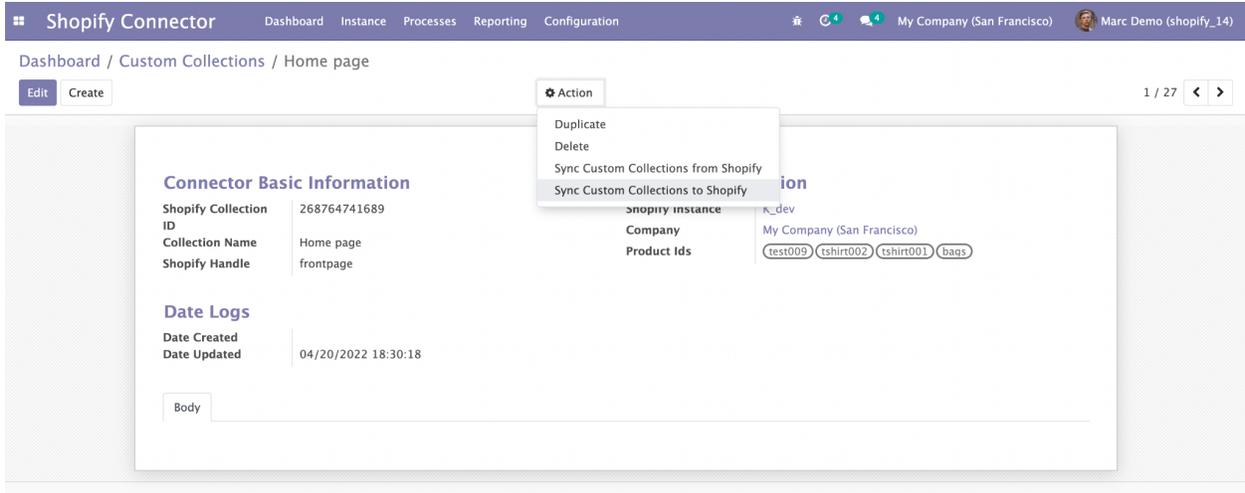
### Instance Information

Shopify Instance: K\_dev

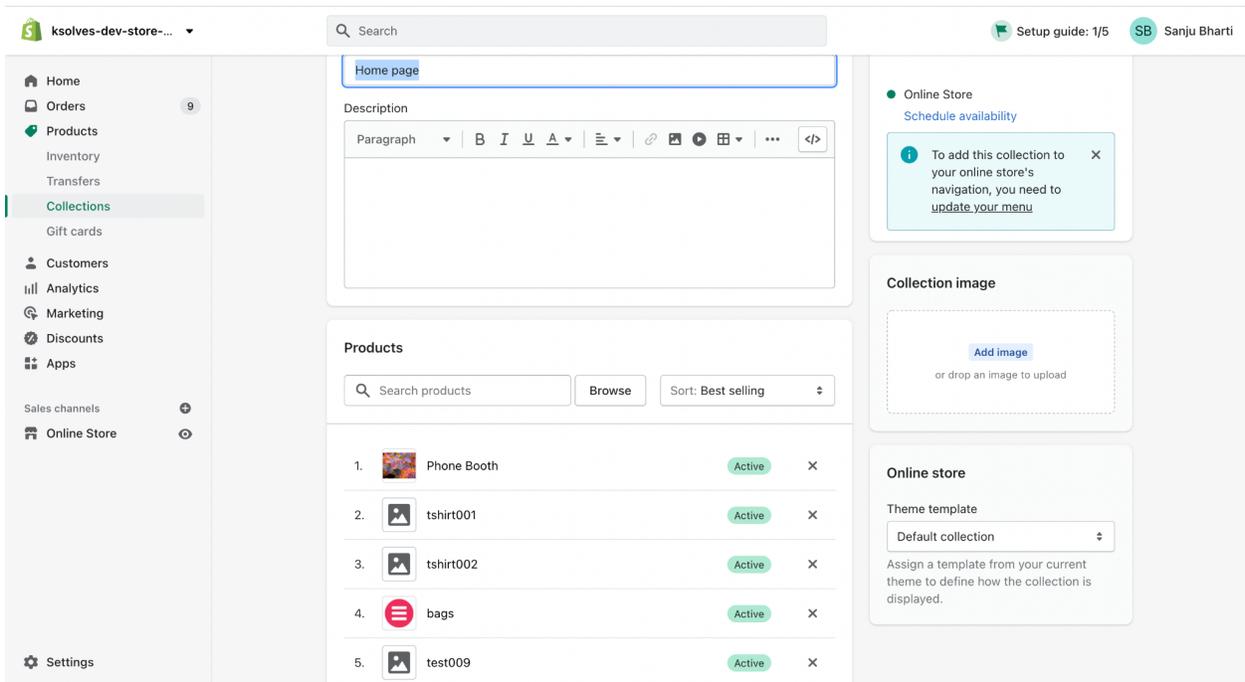
Company: My Company (San Francisco)

Product Ids: test009 tshir002 tshir001 bags

Click on the Action>Sync Custom Collections to Shopify

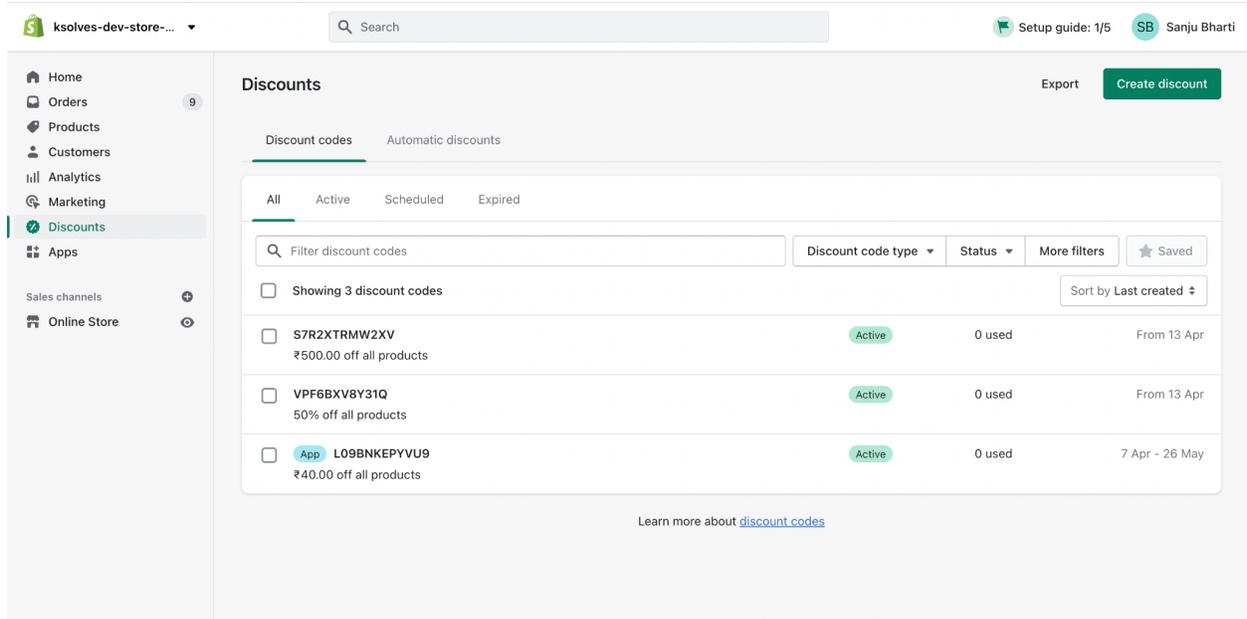


Verify from the Shopify End. Go to Shopify store>Products>Collections>Open your collection. There you will see your exported products associated with a particular collection.



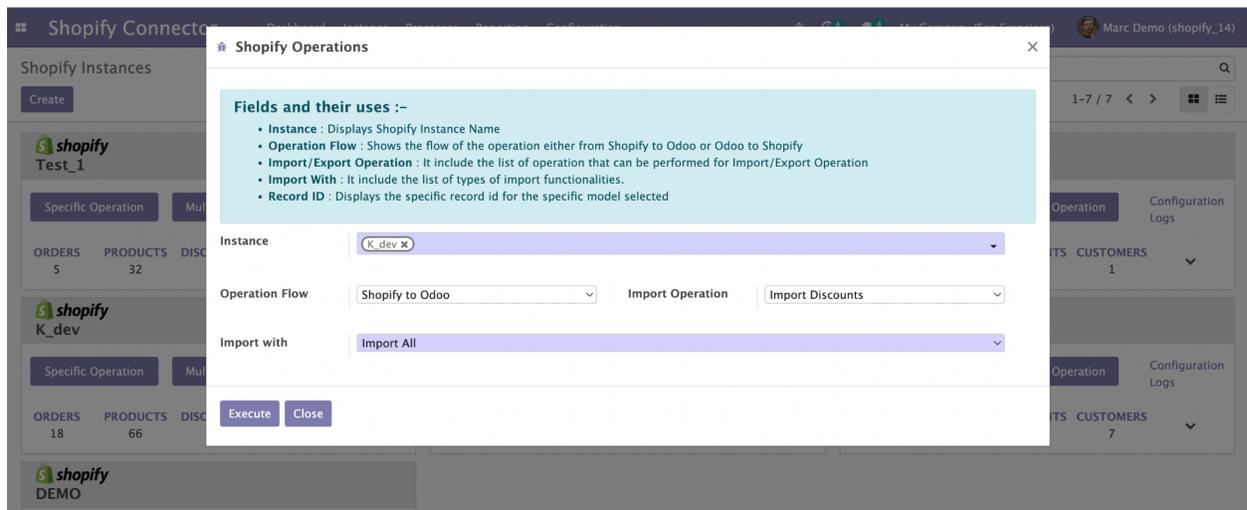
## 4.4.2 Discounts

Create a discount in Shopify. Go to Shopify store>Discounts>Create Discount



Once you create your discount(s) in Shopify go back to the Odoo Shopify Instance.

Import all Discounts:



All the discounts have been listed under the Queue Jobs

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
L098NKEPVU9	967231275065	Discounts	Shopify to Shopify Layer	Import	K_dev	New
VPF68XV8Y31Q	972586418233	Discounts	Shopify to Shopify Layer	Import	K_dev	New
S7R2XTRMW2XV	972590383161	Discounts	Shopify to Shopify Layer	Import	K_dev	New
Home page	268764741689	Collections	Shopify to Shopify Layer	Import	K_dev	Done
dif phase 3	63720161337	Location	Shopify to Shopify Layer	Import	K_dev	Done
Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed
Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Done

## Go to Dashboard > Discounts

Shopify Discount ID	Discount Code	Amount	Shopify Instance	Date Coupon Starts	Date Coupon Expires
972590383161	*****	-500.00	K_dev	04/13/2022 14:20:48	
972586418233	*****	-50.00	K_dev	04/13/2022 13:06:56	
967231275065	*****	-40.00	K_dev	04/08/2022 00:16:51	05/27/2022 00:16:51
953253429292	*****	-100.00	Demo test	01/31/2022 17:28:39	
953237733420	*****	-5.00	Demo test	01/31/2022 13:52:18	
953237569580	*****	-29.00	Demo test	01/31/2022 13:50:38	
953237438508	*****	-100.00	Demo test	01/31/2022 13:49:30	02/06/2022 13:49:49
953227640876	*****	-20.00	Demo test	01/31/2022 13:27:31	
953227542572	*****	-20.00	Demo test	01/31/2022 13:26:05	
953227411500	*****	-25.00	Demo test	01/31/2022 13:26:05	

Either you can create a new discount or export the existing one.

Once you create a discount you can export it to Shopify as shown below:

Dashboard / Discounts / S7R2XTRMW2XV

Button : Export to Shopify

Object: ks.shopify\_discounts  
 Modifiers: [{"ks\_title": "", "invisible": true}]\n\nButton Type: object\n\nMethod: ks\_manage\_shopify\_discounts\_export

**Basic Information**

**S7R2XTRMW2XV**

<b>Amount</b>	-500.00	<b>Instance Details</b>	
<b>Shopify Discount ID</b>	972590383161	<b>Shopify Instance</b>	K_dev
<b>Shopify Price Rule ID</b>	12526214578233	<b>Company</b>	My Company (San Francisco)
<b>Allocation Method</b>	The calculated discount amount will be applied across the entitled items	<b>Date Created</b>	04/13/2022 14:21:13
<b>Date Coupon Starts</b>	04/13/2022 14:20:48	<b>Date Updated</b>	04/13/2022 14:21:13
<b>Date Coupon Expires</b>			

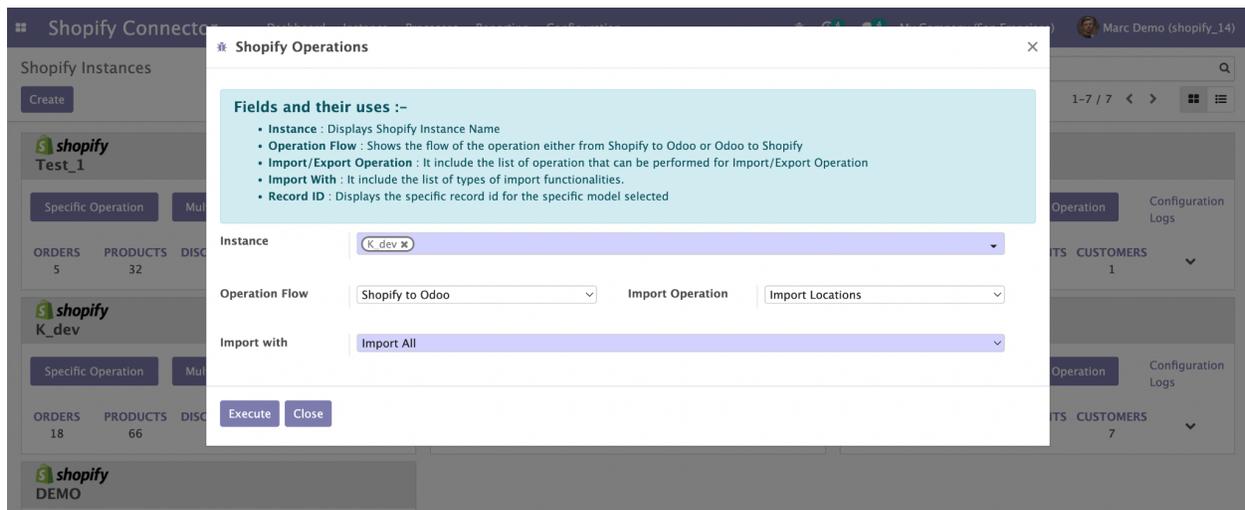
Price Rules Target Settings

### 4.4.3 Locations

Importing location suggests to which location your Shopify store and its Orders and Products are associated.

For instance:

Import All Locations in Odoo for a particular Instance.



Go to Processes>Queue Jobs, and open the Location

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
dlf phase 3	63720161337	Location	Shopify to Shopify Layer	Import	K_dev	New
S7R2XTRMW2XV	972590383161	Discounts	Shopify Layer to Shopify	Export	K_dev	Done
L09BNKEPYU9	967231275065	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
VPF6BXV8Y31Q	972586418233	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
S7R2XTRMW2XV	972590383161	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
Home page	268764741689	Collections	Shopify to Shopify Layer	Import	K_dev	Done

This is the location of your Shopify store which is synced with Odoo. Now, all the products, orders, customers, and more will be synced with this location of your Shopify store.

The screenshot displays the Shopify Connector interface. At the top, there is a navigation bar with 'Shopify Connector' and various menu items like 'Dashboard', 'Instance', 'Processes', 'Reporting', and 'Configuration'. The user is logged in as 'Marc Demo (shopify\_14)'. Below the navigation bar, the breadcrumb trail shows 'All Queue Jobs / dlf phase 3'. The main content area features a 'Process Record' for 'dlf phase 3'. This record is divided into two columns: 'Instance Information' and 'Other Information'. The 'Instance Information' column lists 'Domain' (Location), 'Shopify ID' (63720161337), 'Odoo ID' (0), and 'Shopify Instance' (K\_dev). The 'Other Information' column lists 'Operation Flow' (Shopify to Shopify Layer), 'Operation Performed' (Import), and 'Created on' (04/20/2022 17:11:52). Below these columns, there is a 'Data' section containing a JSON object with detailed instance information.

Instance Information		Other Information	
Domain	Location	Operation Flow	Shopify to Shopify Layer
Shopify ID	63720161337	Operation Performed	Import
Odoo ID	0	Created on	04/20/2022 17:11:52
Shopify Instance	K_dev		

```
{
  "id": "63720161337",
  "name": "dlf phase 3",
  "address1": "dlf phase 3",
  "address2": null,
  "city": "gurugram",
  "zip": "122002",
  "province": "Haryana",
  "country": "IN",
  "phone": "",
  "created_at": "2022-03-07T14:27:43+05:30",
  "updated_at": "2022-03-07T14:27:43+05:30",
  "country_code": "IN",
  "country_name": "India",
  "province_code": "HR",
  "legacy": false,
  "active": true,
  "admin_graphql_api_id": "gid://shopify/Location/63720161337",
  "localized_country_name": "India",
  "localized_province_name": "Haryana"
}
```

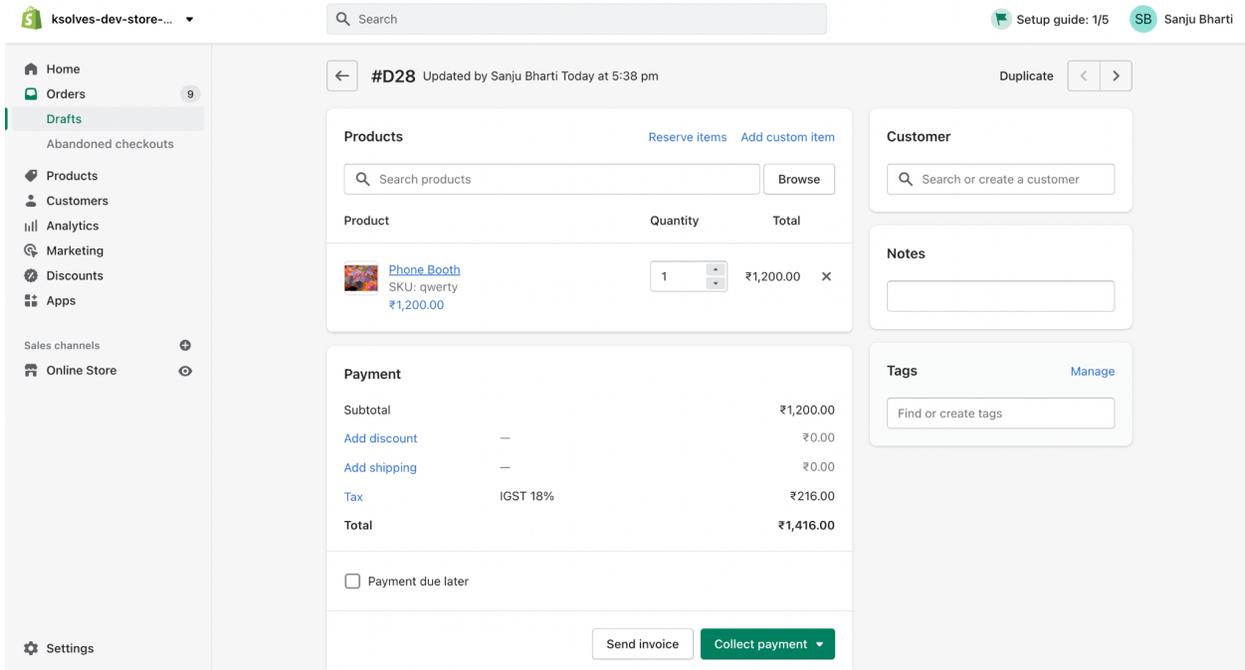
#### 4.4.4 Draft Orders

Draft orders are orders that are created by a Merchant on the behalf of a customer.

Draft Order is converted into order when a Merchant gets payment for it and generates an invoice.

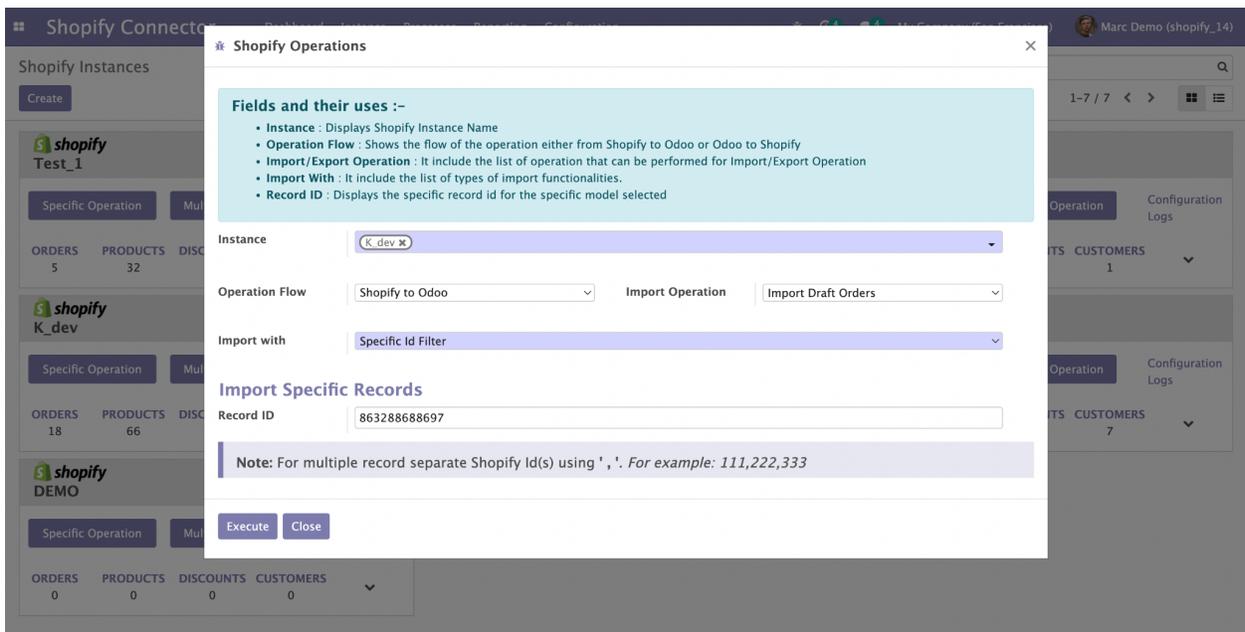
If you have draft orders in Shopify you can easily import the data into Odoo.

Go to Shopify>Orders>Draft Orders>Create a draft order



Copy the unique Id of the draft order and go to Odoo Shopify Connector.

Instance>Specific Operations>Import Draft Order with Record Id



## Verify from the Processes>Queue Jobs

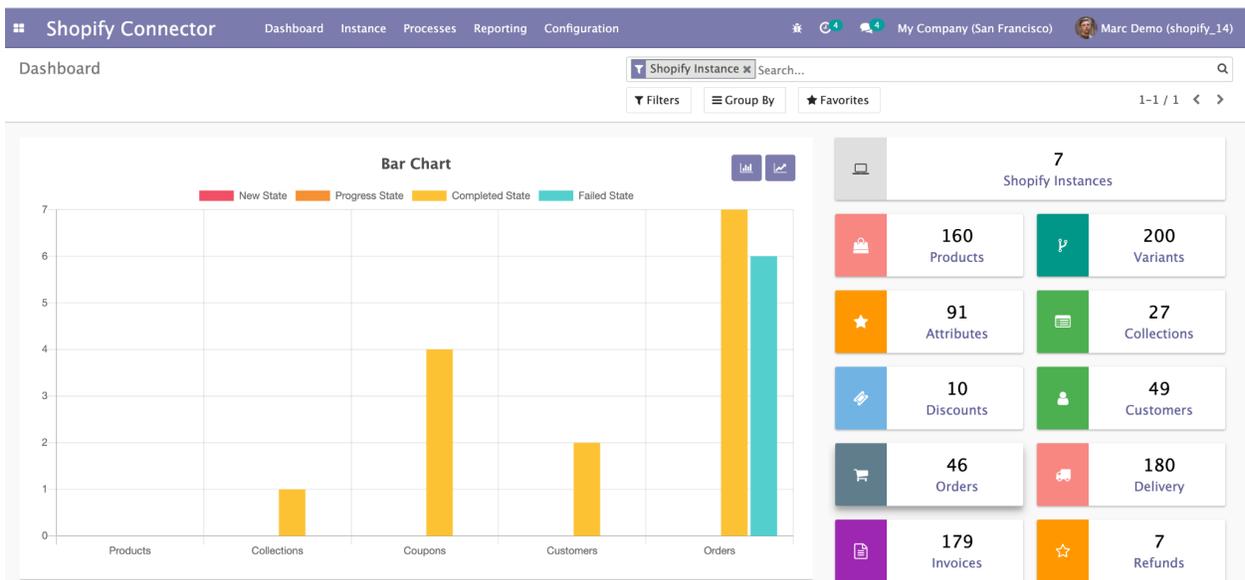
Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

All Queue Jobs Search...

Filters Group By Favorites 1-24 / 24 < >

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/> Sale Order #863288688697	863288688697	Sale Order	Shopify To Odoo	Import	K_dev	New
<input type="checkbox"/> dlf phase 3	63720161337	Location	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> S7R2XTRMW2XV	972590383161	Discounts	Shopify Layer to Shopify	Export	K_dev	Done
<input type="checkbox"/> L09BNKEPYVU9	967231275065	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> VPF6BXV8Y31Q	972586418233	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> S7R2XTRMW2XV	972590383161	Discounts	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Home page	268764741689	Collections	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> dlf phase 3	63720161337	Location	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed

## Go to Dashboard>Orders>Click on your draft order



Here you will notice that the draft order is created under a Guest Customer because we did not add any customer while creating the draft order in Shopify.

You can add a customer while creating a draft order in Shopify.

If you need to export a draft order you need to note a few things. Click on the Shopify Info tab and check the 'Is Draft Order' box. Click on the Action tab and choose the Export order in Shopify.

**Fields and their uses :-**

- Instance : Displays Shopify Instance Name
- Shopify Id : Displays Shopify ID
- Created On : Displays date of creation
- Updated On : Displays date of updation
- Shopify Status : Displays Shopify Order Status
- Shopify Payment Gateway : Displays Shopify Order Payment Gateway
- Transaction Id : Displays Shopify Order Transaction ID
- Customer IP : Displays Shopify Customer IP
- Shopify Coupons : Displays Shopify Order Coupons

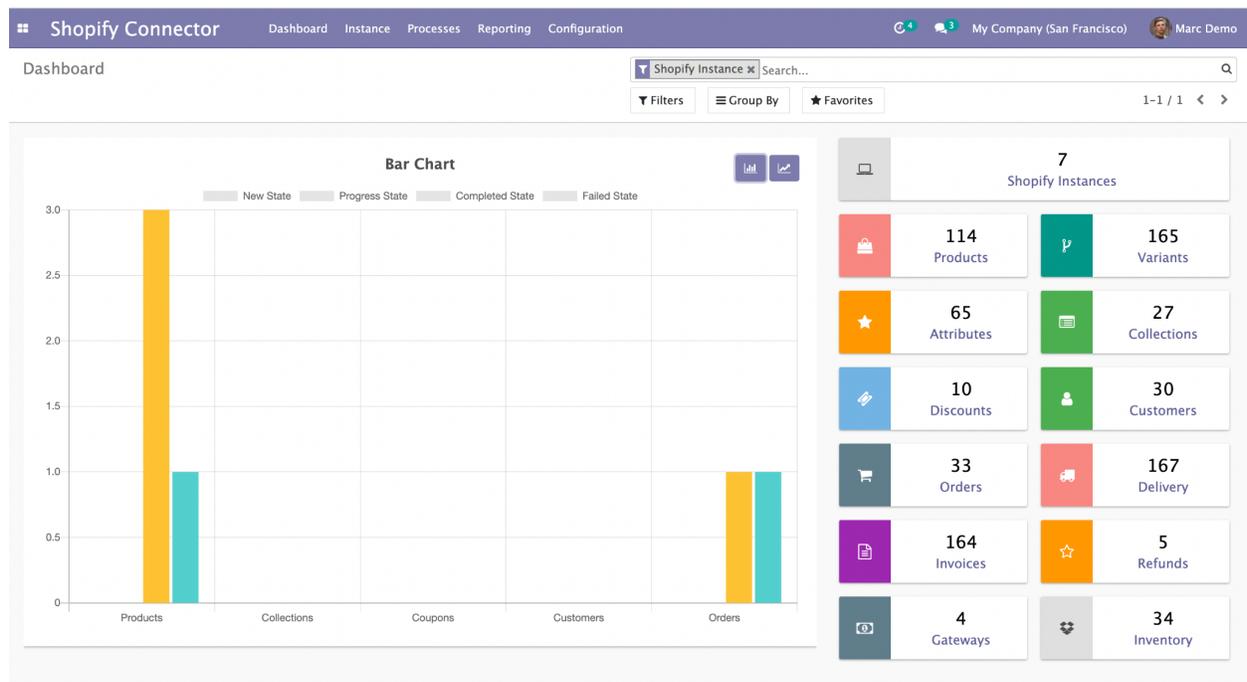
Instance	K_dev	Shopify Status	Open
Shopify Id	863288688697	Shopify Payment Gateway	
Shopify Draft Id	0	Transaction Id	
Is Draft Order?	<input checked="" type="checkbox"/>	Checkout Id	
Created On	04/20/2022 19:38:19	Customer IP	
Updated On	04/20/2022 19:38:19		

Shopify Coupons

## 5. Dashboard

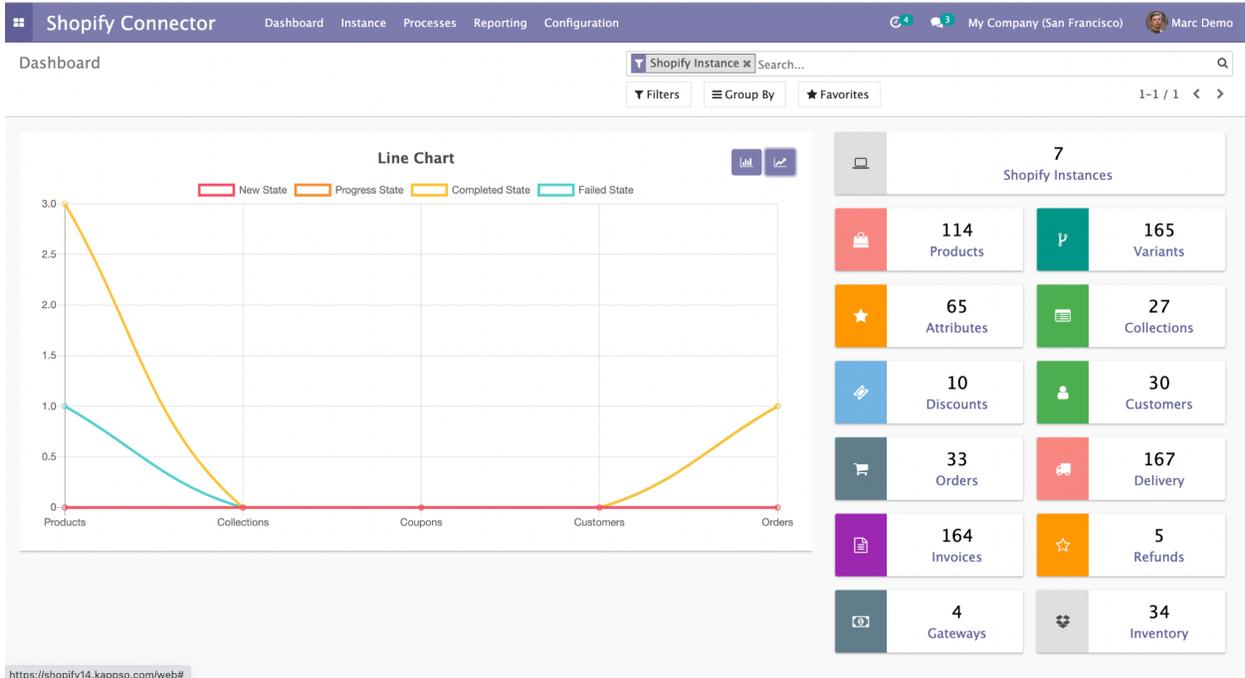
The dashboard has two sections, one is a graphical representation of all the records that are synced using the Shopify Connector for each day. The other includes the total number of Shopify instances, Products, Attributes, Collections, Customers, Discounts, and more.

It is a quick way to highlight the data in the form of a Bar Chart and Line Chart.



Here you can see the New State, Progress State, Completed State, and Failed States of your records.

The bar chart can be converted to the Line Chart. These graphical representations are helpful if you need to give a presentation about your syncing performed using the instance on daily basis.



If you click over any record, say Product, you will find the entire product list, which includes the Shopify ID, Odoo Product Template, Instance, Date Created, and Date Updated to that of a particular product.

The screenshot shows the 'Shopify Connector' dashboard with the 'Shopify Product Template' view. It displays a table with the following columns: Shopify Product ID, Odoo Product Template, Instance, Date Created, and Date Updated. The table contains 20 rows of data.

Shopify Product ID	Odoo Product Template	Instance	Date Created	Date Updated
6695876034617	1_image_test	Test	03/09/2022 14:09:44	04/15/2022 21:32:21
6729172713529	Steel Bottles	K_dev	04/15/2022 21:23:03	04/15/2022 21:23:03
6729134473273	[painting] Paintings	Test	04/15/2022 20:42:14	04/15/2022 21:04:38
6693668913209	[ASDF1234567] A-1	Test	03/07/2022 21:59:54	04/05/2022 15:33:26
6693668913209	[ASDF1234567] A-1	K_dev	03/08/2022 02:29:54	04/05/2022 19:03:26
6702461681721	A-Poco 004	K_dev	03/15/2022 05:06:59	03/15/2022 05:06:59
6697901523001	Conference Chair (CONFIG)	K_dev	03/11/2022 01:28:11	03/11/2022 01:28:19
6697901555769	Conference Chair (CONFIG)	K_dev	03/11/2022 01:28:12	03/11/2022 02:24:00
6706940346425	ky5	K_dev	03/22/2022 19:37:40	03/22/2022 19:37:42
6695982563385	test009	K_dev	03/09/2022 19:51:06	03/09/2022 19:51:06
6726927941689	U1002	K_dev	04/14/2022 17:44:49	04/14/2022 17:44:50
6702460862521	A-Poco 004	K_dev	03/15/2022 04:59:33	03/15/2022 04:59:34
6695876034617	1_image_test	K_dev	03/09/2022 14:09:44	04/15/2022 21:32:21
6702460239929	A-Poco 004	K_dev	03/15/2022 04:54:33	03/15/2022 04:54:34
6695875838009	5A	K_dev	03/09/2022 18:39:39	03/23/2022 16:24:39
6695875772473	6-A	K_dev	03/09/2022 18:39:37	03/09/2022 18:48:56
6726924075065	U1001	K_dev	04/14/2022 14:13:02	04/14/2022 14:13:03
6699640717369	Redmi 01	K_dev	03/11/2022 19:26:58	04/14/2022 13:59:30
6726885572665	[jkl] Stands	K_dev	04/14/2022 13:41:55	04/14/2022 13:55:56
6724630806585	Jeth Sweatshirt in Blue/Royal	K_dev	04/13/2022 14:05:59	04/13/2022 14:05:59

Moreover, we have given a Filter here to sort your product based on the 'Today Created/Updated' option.

Shopify Product ID	Odoo Product Template	Instance	Today Created Shopify Products	Date Created	Date Updated
6730789126201	[Tee-1] Red T-shirt	K_dev	Today Updated Shopify Products	03/18/2022 18:22:00	04/18/2022 19:01:15
6695875903545	1_image_test_01	K_dev	Add Custom Filter	03/22/2022 14:09:41	03/09/2022 14:09:41
6702086979641	A-Poco 001	K_dev 1		03/14/2022 15:21:54	03/23/2022 09:56:41
6702148452409	A-Poco 002	K_dev 1		03/14/2022 16:54:16	03/14/2022 17:17:29
6702446149689	A-Poco 003	K_dev 1		03/15/2022 00:07:21	03/15/2022 00:07:22
6702450245689	A-Poco 003	K_dev 1		03/15/2022 00:09:36	03/15/2022 00:09:37
6697934585913	Apple	K_dev 1		03/10/2022 21:42:35	03/10/2022 21:42:36
6706958008377	bags	K_dev 1		03/22/2022 16:47:54	03/22/2022 17:03:23
6707019350073	chair123	K_dev 1		03/22/2022 17:03:07	03/22/2022 17:03:18
6707489833017	cooler	K_dev 1		03/23/2022 13:02:06	03/23/2022 13:02:08

To quickly check all the details associated with a particular product, click on it. A window containing all the product details will appear.

Shopify Connector
Dashboard Instance Processes Reporting Configuration
My Company (San Francisco) Marc Demo

Dashboard / Shopify Product Template / Steel Bottles
2 / 80

Edit Create
Action

**Fields and their uses :-**

- **Odoo Product Template** : Displays Odoo Linked Product Template Name
- **Shopify Product Type** : Displays Shopify Product Type
- **Shopify Product ID** : Displays Shopify ID
- **Instance** : Displays Shopify Instance Name
- **Date Created** : Displays date of creation
- **Date Updated** : Displays date of updation
- **Tags** : Displays Shopify Tags
- **Shopify Regular Price** : Displays Shopify Regular Price
- **Shopify Sale Price** : Displays Shopify Sale Price

<p>Odoo Product Template: Steel Bottles</p> <p>Shopify Product Type: Simple Product</p> <p>Product Type:</p> <p>Tags:</p> <p>Vendor: ksolves-dev-store-001</p> <p>Inventory Policy: Deny</p>	<p>Shopify Product ID: 6729172713529</p> <p>Collection Ids:</p> <p>Shopify Product Active Status: <input type="checkbox"/></p> <p>Instance: K_dev</p> <p>Inventory ID: 42179012034617</p> <p>Date Created: 04/15/2022 21:23:03</p> <p>Date Updated: 04/15/2022 21:23:03</p>	<p>Shopify Regular Price: 1,500.00 →</p> <p>Shopify Compare Price: 0.00 →</p>
--	---	---

General
Gallery Images

**Description**

This is a time-saving option for tracking the details of any record instantly through the dashboard and saving multiple clicks.

## 6. Meta Mapping

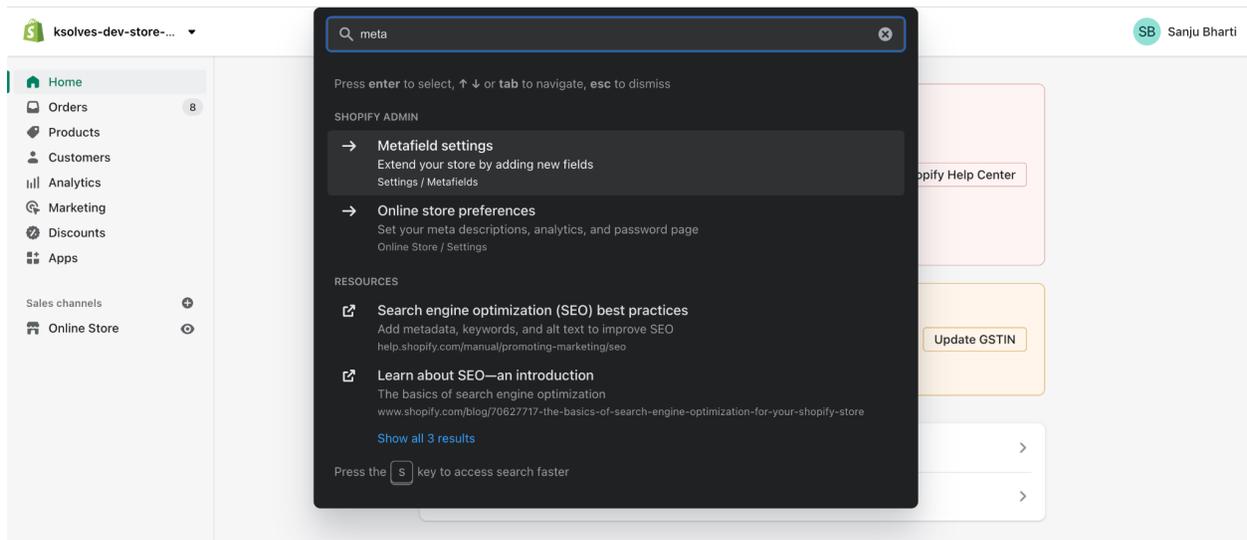
Whenever we are importing a record from Shopify to Odoo, there are certain data associated with a record, which does not get synced. To connect one set of metadata fields to their corresponding elements in systems, the Meta Mapping feature is used.

Here, a metadata field from Shopify will be mapped with fields of Odoo.

**Note:** Meta mapping only works in the case of the Import function.

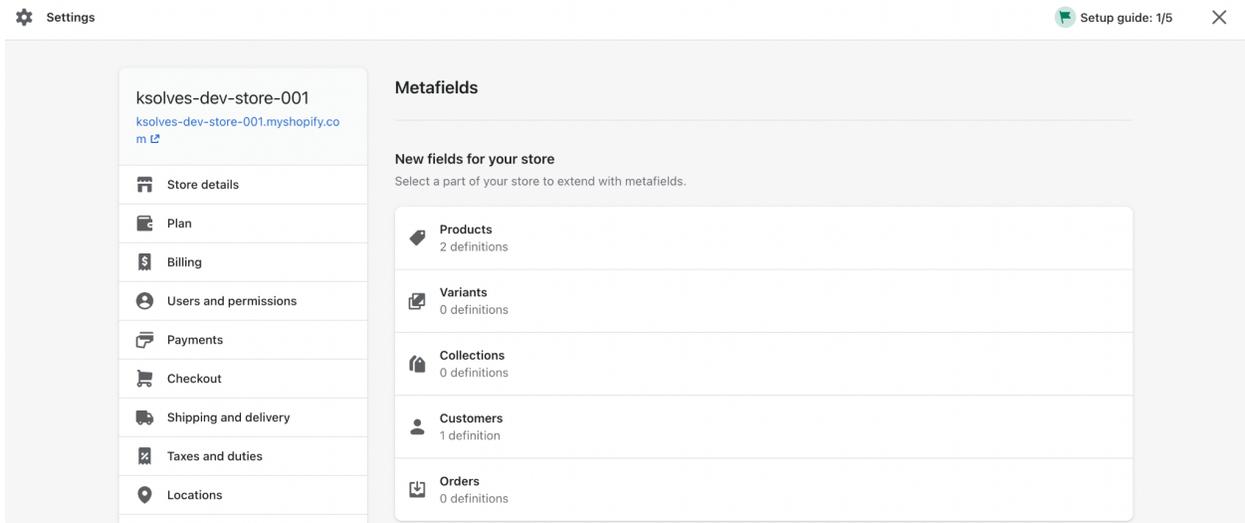
Let's begin with how we can map the data from Shopify to Odoo.

Go to Shopify backend and search for Meta field settings.

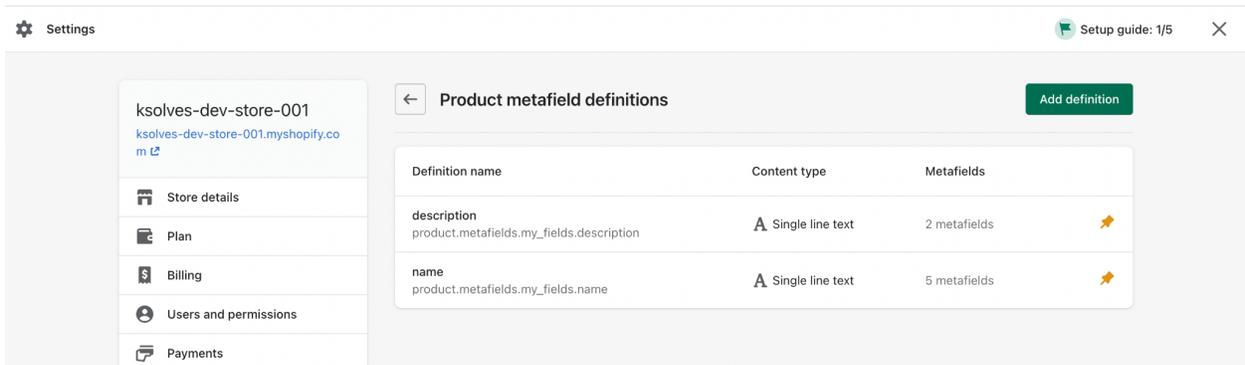


The Metafields are associated with Products, Variants, Collections, Customers, and Others.

You can choose any of the meta fields from the given list.



We have chosen the Product Metafields. Click on Add Definition button.



Now, add product meta fields definition as much as you require.

For example, if you want to add a meta field for Size, then add all the required details.

Unsaved changes Setup guide: 1/5 Discard Save

ksolves-dev-store-001  
[ksolves-dev-store-001.myshopify.com](https://ksolves-dev-store-001.myshopify.com)

- Store details
- Plan
- Billing
- Users and permissions
- Payments
- Checkout
- Shipping and delivery
- Taxes and duties
- Locations

### Add product metafield definition

Definition pinned

Name: Size

Namespace and key: my\_fields.size\_

Description: Small Size 10/100

Expose this metafield to Storefront API requests. [Learn more](#)  
Give read access to apps, storefronts, or anything using the Storefront API.

[+ Select content type](#)

Note: It is mandatory to select the content type. After filling in all the data, click on the Save button.

Unsaved changes Setup guide: 1/5 Discard Save

ksolves-dev-store-001  
[ksolves-dev-store-001.myshopify.com](https://ksolves-dev-store-001.myshopify.com)

- Store details
- Plan
- Billing
- Users and permissions
- Payments
- Checkout
- Shipping and delivery
- Taxes and duties
- Locations
- Gift cards
- Markets

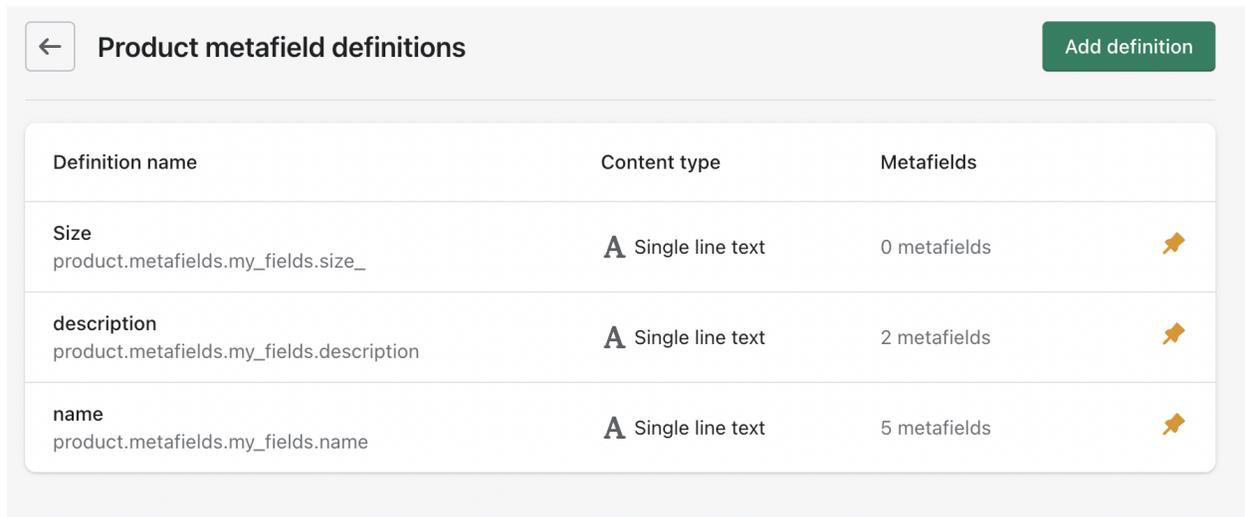
### Add product metafield definition

Definition pinned

- Color
- Date and time
- File
- JSON
- Measurement
- Number
- Rating
- Reference
- Text**
- True or false
- URL

[+ Select content type](#)

Here we have given three meta field definitions: Size, Description, and Name.

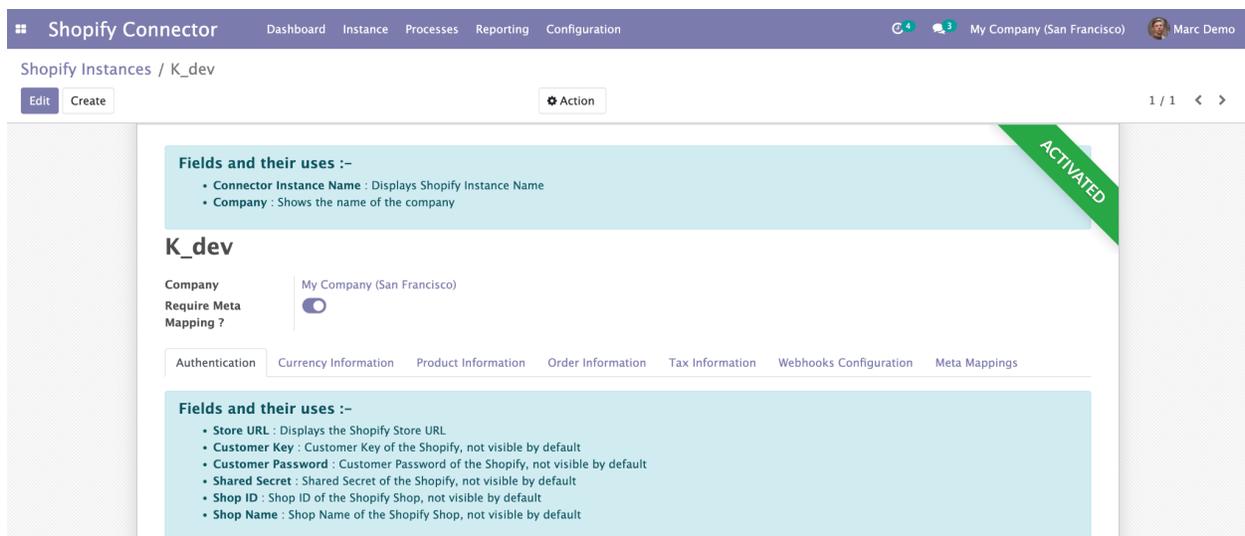


The screenshot shows a table titled "Product metafield definitions" with a back arrow on the left and an "Add definition" button on the right. The table has three columns: "Definition name", "Content type", and "Metafields".

Definition name	Content type	Metafields
Size product.metafields.my_fields.size_	A Single line text	0 metafields
description product.metafields.my_fields.description	A Single line text	2 metafields
name product.metafields.my_fields.name	A Single line text	5 metafields

Now, go to Odo Instance. There you will see the option 'Require Meta Mapping?'.

As you turn it ON, you will see a Meta Mappings tab appeared to the extreme right created.



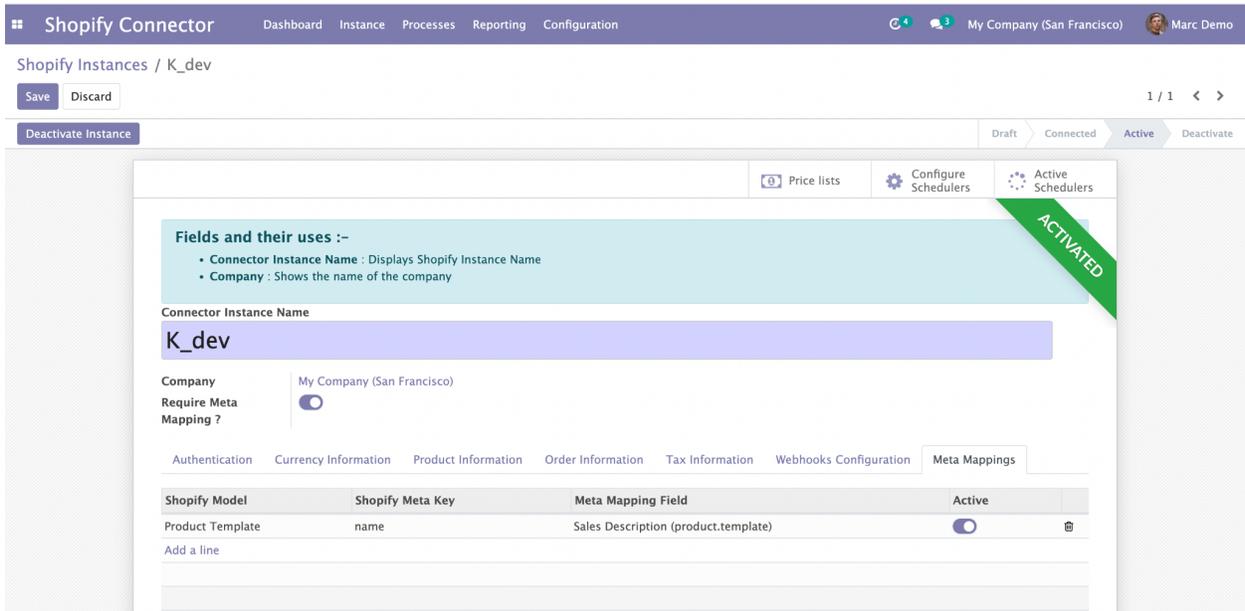
The screenshot shows the "Shopify Connector" configuration page for an instance named "K\_dev". The page has a top navigation bar with "Shopify Connector", "Dashboard", "Instance", "Processes", "Reporting", and "Configuration". The user is logged in as "Marc Demo" from "My Company (San Francisco)".

The main content area shows the configuration for "K\_dev". There is a "Company" field set to "My Company (San Francisco)". The "Require Meta Mapping?" toggle is turned ON, and a green "ACTIVATED" banner is visible in the top right corner. Below this, there are tabs for "Authentication", "Currency Information", "Product Information", "Order Information", "Tax Information", "Webhooks Configuration", and "Meta Mappings".

Under the "Meta Mappings" tab, there is a section titled "Fields and their uses :-" with the following list:

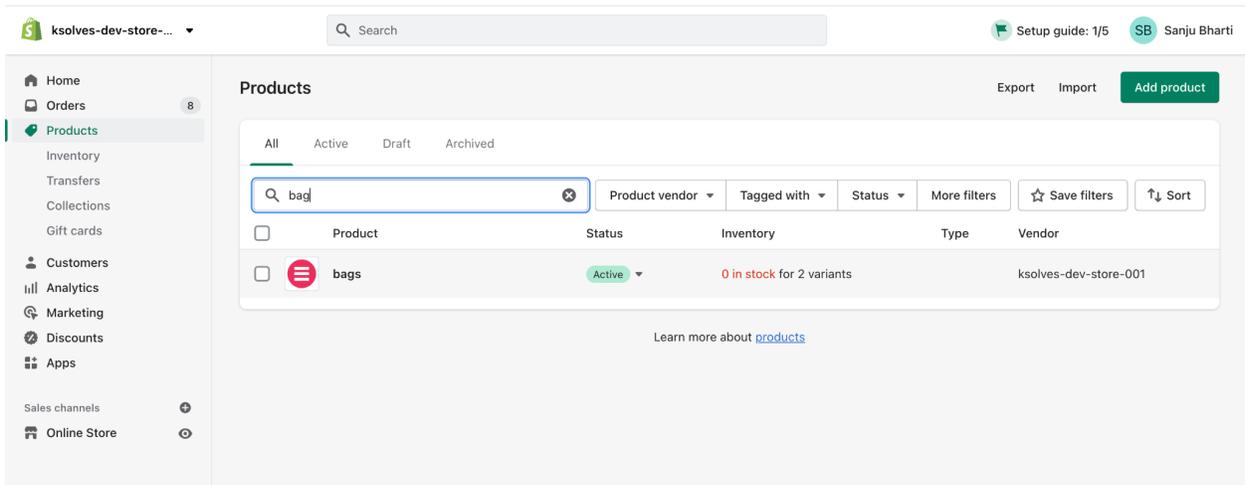
- Store URL : Displays the Shopify Store URL
- Customer Key : Customer Key of the Shopify, not visible by default
- Customer Password : Customer Password of the Shopify, not visible by default
- Shared Secret : Shared Secret of the Shopify, not visible by default
- Shop ID : Shop ID of the Shopify Shop, not visible by default
- Shop Name : Shop Name of the Shopify Shop, not visible by default

Edit the Meta field, give Shopify Model, Meta Key, and Active the toggle button. Click on the Save button.

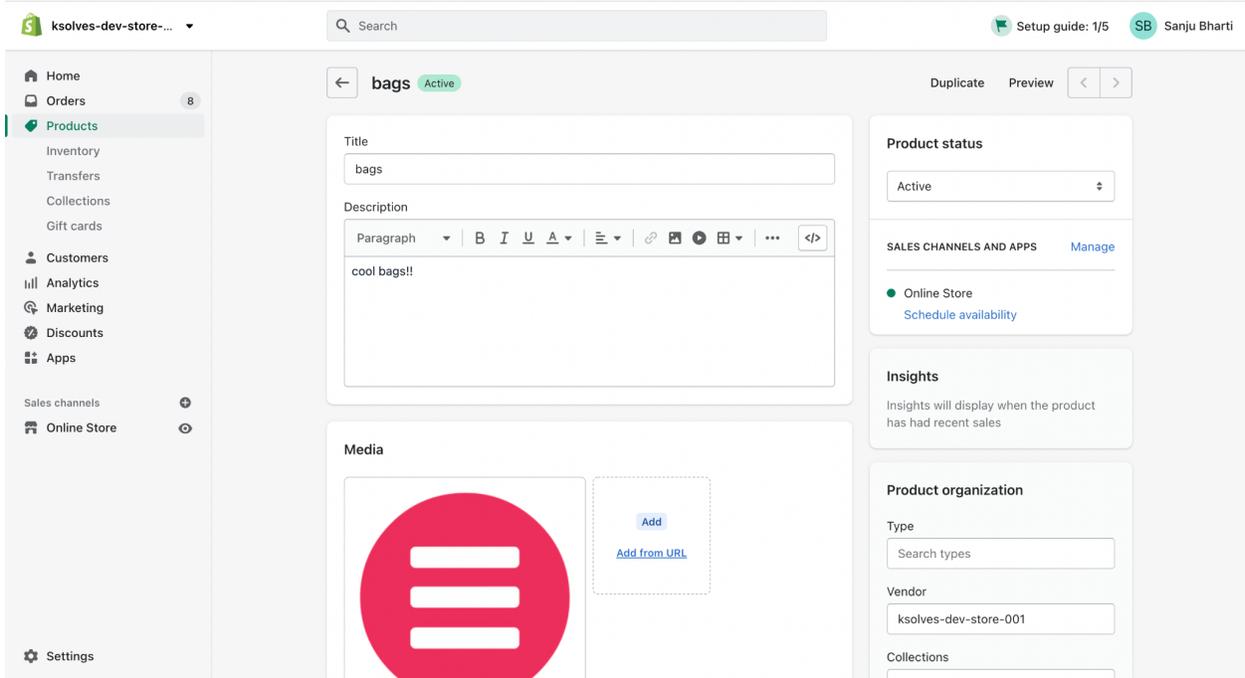


Here, what happens is that the name Meta field of the product will be mapped with the Sale Description in Odoo.

Go to the Shopify backend and find a product whose meta fields you need to map with Odoo.

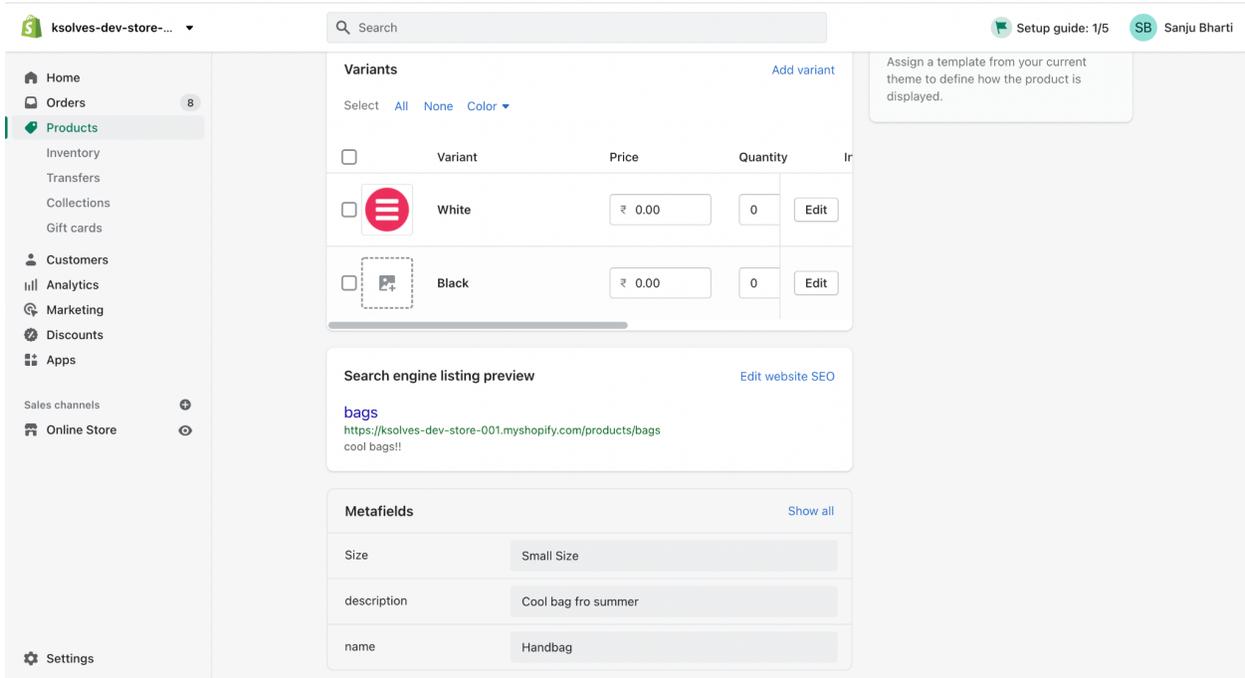


You can either choose the already created product or create a new one.

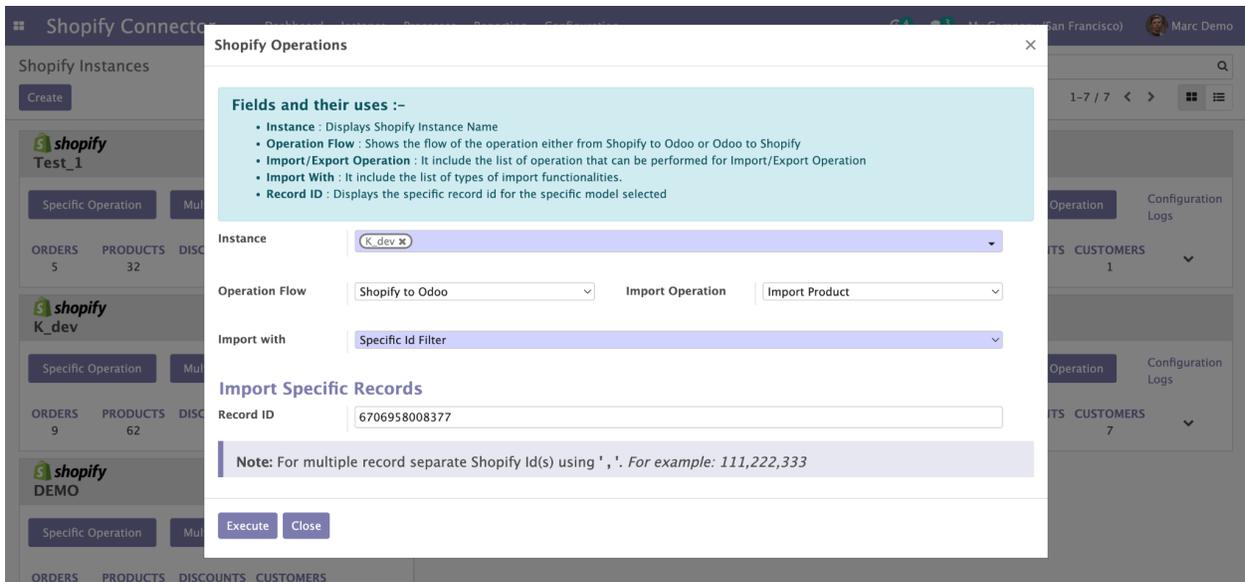


You can edit the product Metafields if not mentioned and then click on the Save button.

Copy the unique Id of the product for performing import operations.



Now, go back to Odoo Instance>Specific Operation>Import the product using its record Id. Click Execute. The product meta fields will be imported.



To verify, go to Processes>Queue Jobs. Here you will see the product record to be imported, with the associated Instance.

Shopify Connector Dashboard Instance Processes Reporting Configuration Marc Demo

All Queue Jobs Search...

Filters Group By Favorites 1-5 / 5

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
bags	6706958008377	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
bags	6706958008377	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
bags	6706958008377	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
Pen holder	6731024924729	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
Pen holder	6731024924729	Product Template	Shopify to Shopify Layer	Import	K_dev	Done

To check the meta fields, go to Sales>Products>Products

Sales Orders To Invoice Products Reporting Configuration Marc Demo

Quotations My Quotations Search...

Create Filters Group By Favorites 1-28 / 28

**Company Data**

Set your company's data for documents header/footer.

Let's start!

**Quotation Layout**

Customize the look of your quotations.

★ Looks great!

**Order Confirmation**

Choose between electronic signatures or online payments.

Set payments

**Sample Quotation**

Send a quotation to test the customer portal.

Send sample

Number	Creation Date	Customer	Salesperson	Next Activity	Company	Total	Status
SC #1086670340321	03/28/2022	gomez	Marc Demo		My Company (San Francisco)	\$ 100.00	Sales Order
SC #4685408174305	03/07/2022	lovely	Marc Demo		My Company (San Francisco)	\$ 1.00	Sales Order
S00042	03/07/2022	lovely sr	Marc Demo		My Company (San Francisco)	\$ 1.15	Quotation
S00041	01/31/2022	test_customer_1	Marc Demo		My Company (San Francisco)	\$ 0.00	Quotation
S00038	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 116.15	Quotation
S00036	12/06/2021	Test Cust	Marc Demo		My Company (San Francisco)	\$ 1.15	Cancelled
S00031	10/06/2021	Farooq Nasim	Marc Demo		My Company (San Francisco)	\$ 115,000.00	Quotation
S00028	10/06/2021	Gaurav Kumar	Marc Demo		My Company (San Francisco)	\$ 17,250.00	Sales Order

Search for your product and open it.

Sales Orders To Invoice Products Reporting Configuration Marc Demo

Products Product bags Search...

Create Filters Group By Favorites 1-2 / 2

bags

2 Variants

Price: \$ 1.00

On hand: 0.00 Units

bags

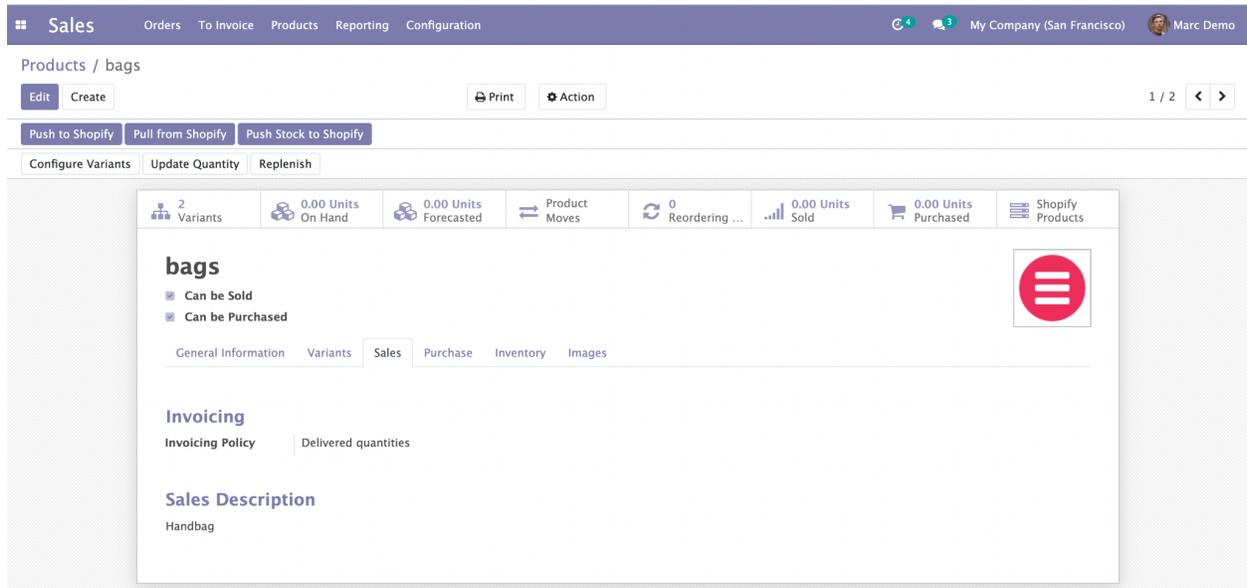
2 Variants

Price: \$ 1.00

On hand: 0.00 Units

In the Sales tab, you will see that the name of the product has been mapped with the Sales Description field in Odoo.

This means the 'Handbag' name that we mentioned in Shopify is mapped with the Odoo Sales Meta field.



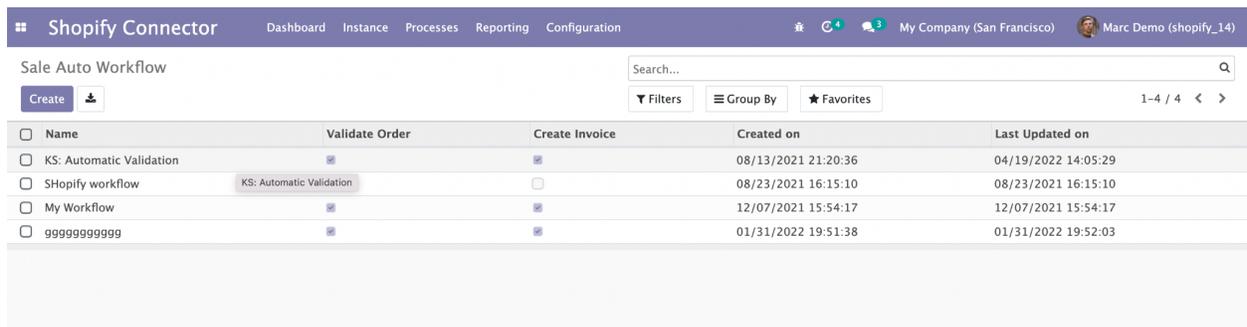
You can Map any specific data from Shopify to Odoo in a similar way!

## 7. Sale Workflow

The order cycle on Shopify and Odoo is different. To update the Order Status in Odoo, we use the Sale Workflow function, which is an automatic operation once it is configured manually.

Go to Odoo Shopify Connector>Configuration>Sale Workflow

You will see a window like this:



You can create your Sale Workflow or choose the one that is already existed. We have created the KS: Automatic Validation. If we click on it, it will appear like this:

The screenshot shows the 'Shopify Connector' interface. The top navigation bar includes 'Dashboard', 'Instance', 'Processes', 'Reporting', and 'Configuration'. The main header displays 'Sale Auto Workflow / KS: Automatic Validation'. Below this, there are 'Edit' and 'Create' buttons, and an 'Action' button. A section titled 'Fields and their uses :-' lists several configuration options with their descriptions. Below this is a configuration form for 'KS: Automatic Validation' with fields for 'Payment Journal', 'Debit Method', 'Sales Journal', 'Bank', and 'Customer Invoices'. A 'Workflow Option' section contains five toggle switches, all of which are turned on.

**Fields and their uses :-**

- **Name** : Sale Auto-Workflow name
- **Sales Journal** : Displays odoo sales journal
- **Validate Order** : If enabled, validates the Shopify order on the basis of given conditions
- **Create Invoice** : If enabled, creates invoice for the Order on the basis of given conditions
- **Confirm Shipment** : If enabled, confirms shipment for the order
- **Register Payment** : If enabled, auto register payment for the order
- **Payment Journal** : Displays the payment journals
- **Debit Method** : Displays method of collecting money
- **Force Invoice Date** : If enabled, force invoice date to be set for today

Name	KS: Automatic Validation
Payment Journal	Bank
Debit Method	Manual
Sales Journal	Customer Invoices

**Workflow Option**

- Validate Order
- Create Invoice
- Register Payment
- Force Invoice Date
- Confirm Shipment

Based on your requirements, you can activate or deactivate the workflow option. We have activated all the 5 workflow options given here.

- **Validate Order:** When a sales order is imported from Shopify, it will be auto-confirmed.
- **Create Invoice:** After the Delivery Order is executed, it will create an invoice automatically.
- **Register Payment:** Once the Invoice has been fully processed, the Payment will be instantly registered in Odoo.
- **Force Invoice Date:** When the Invoice is generated, it will compel the invoice date to be the same as the Sales Order date.
- **Confirm Shipment:** It will confirm the shipment status of the Shopify order in Odoo once imported.

Now, we will go to our Shopify Instance and navigate to the Order Information tab.

It is necessary to add the sale workflow for your order status. You can create a sale workflow as mentioned below:

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Shopify Instances / K\_dev

Edit Create Action 1 / 1 < >

Authentication Currency Information Product Information **Order Information** Tax Information Webhooks Configuration Meta Mappings

**Fields and their uses :-**

- **Default Order Prefix** : Enables/disables – default order prefix of the Odoo
- **Order Prefix** : Prefix added on orders imported from shopify to odoo is shown here
- **Order Status** : Shows the configuration of the imported order status
- **Sales Team** : Shows the location of the sales team
- **Warehouse** : Shows the default warehouse for the stocks
- **Sales Person** : Displays the name of the sales person
- **Import Orders Through** : Shows the status of orders imported portal
- **Payment Terms** : Shows the duration of the payment required to get completed
- **Sale Workflow Configurations** : Shows the flow of the order when imported to the odoo
- **Emails Configurations** : Sends a monthly sales report to the email-ids provided in it.

Default Order Prefix  SC Warehouse San Francisco  
 Order Prefix SC Sales Team Europe  
 Order Status  Open  Paid  Completed  Pending Sales Person Mitchell Admin  
 Import Orders through Status Payment Term

**Sale Workflow Configurations**

Sale Workflow	Order Status
KS: Automatic Validation	<input type="radio"/> Open <input type="radio"/> Paid <input type="radio"/> Completed <input type="radio"/> Pending

Now, create a new order in Shopify, refer to section (4.1 c)

Once you create the order copy it's a unique id.

**Note:** it is completely up to you if you want to fulfill the order or not.

https://ksolves-dev-store-001.myshopify.com/admin/orders/4015694053433?newlyCreated=true

Gmail YouTube Maps Meet - Ksolves M... Odoo developmen... Planet Odoo | Offi... Grammarly Projects - Odoo

ksolves-dev-store-... Search Setup guide: 1/5 SB Sanju Bharti

Home Orders 10 Drafts Abandoned checkouts Products Customers Analytics Marketing Discounts Apps Sales channels Online Store

#1026 Paid Unfulfilled Refund Edit More actions < >

19 April 2022 at 3:33 pm from Draft Orders

**Unfulfilled (1)**

1 Pen holder ₹120.00 × 1 ₹120.00

SKU: holder [Fulfill item](#)

**Paid**

Subtotal	1 item	₹120.00
Tax	IGST 18%	₹21.60
<b>Total</b>		<b>₹141.60</b>
Paid by customer		₹141.60

**Notes** [Edit](#)

No notes from customer

**Customer** [Parul Gautam](#) 4 orders Customer

**CONTACT INFORMATION** [Edit](#)

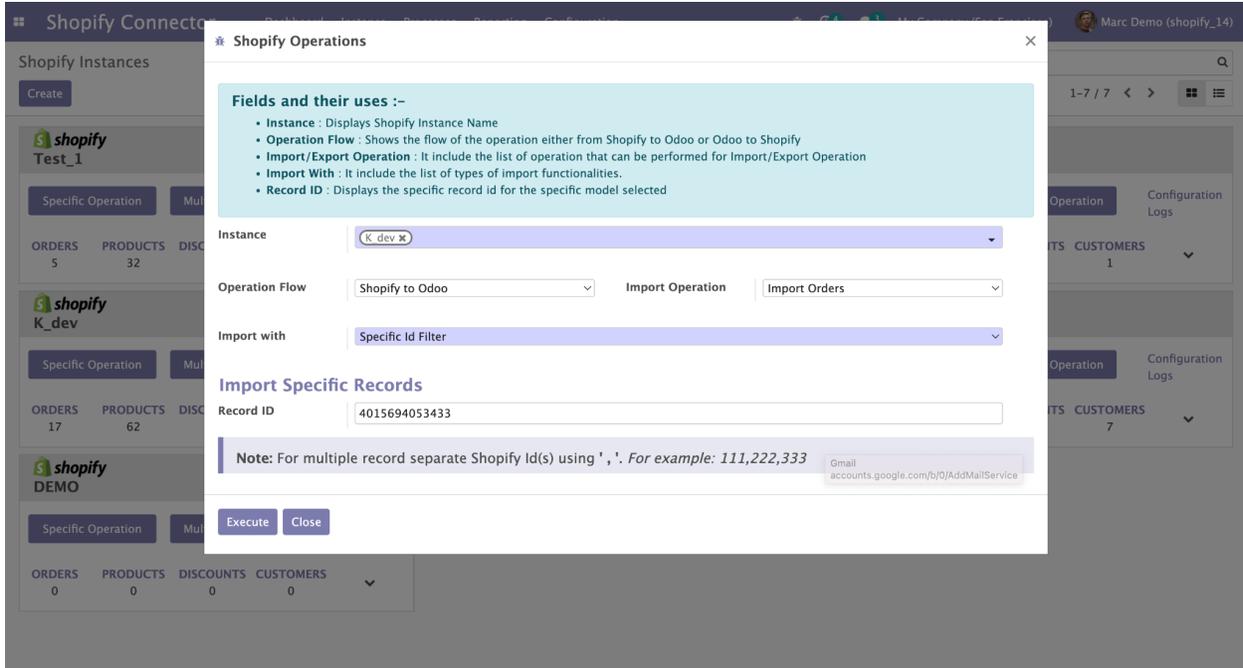
parul.gautam@gmail.com +91 3482 656 214

**SHIPPING ADDRESS** [Edit](#)

Parul Gautam New Delhi 12348 New Delhi DL India +913482656214

**Timeline**  Show comments

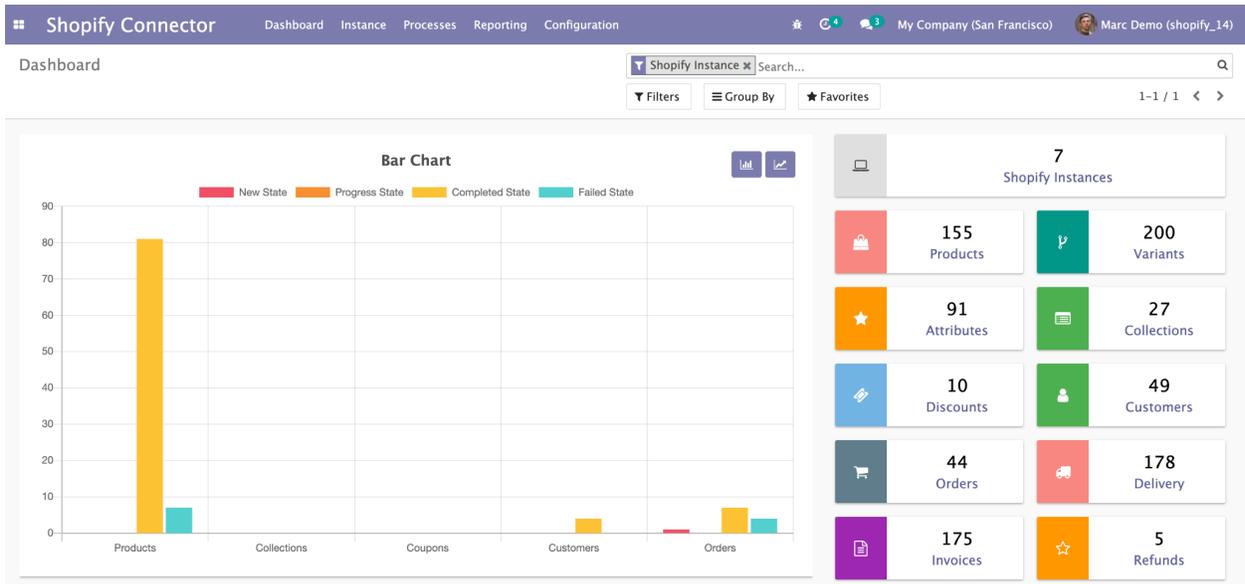
Go to Shopify Instance>Specific Operation>import order with record Id and click on the Execute button.



To check the import order status, go to Processes>Queue Jobs

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	New
<input type="checkbox"/> Pen holder	6731024924729	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Pen holder	6731024924729	Product Template	Shopify to Shopify Layer	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015660662841	4015660662841	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> ky6	6706940543033	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> ky6	6706940543033	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> post it	6706917769273	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> ky7	6706943721529	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> ky7	6706943721529	Product Template	Shopify to Shopify Layer	Import	K_dev	Done

Refer to Dashboard>Orders

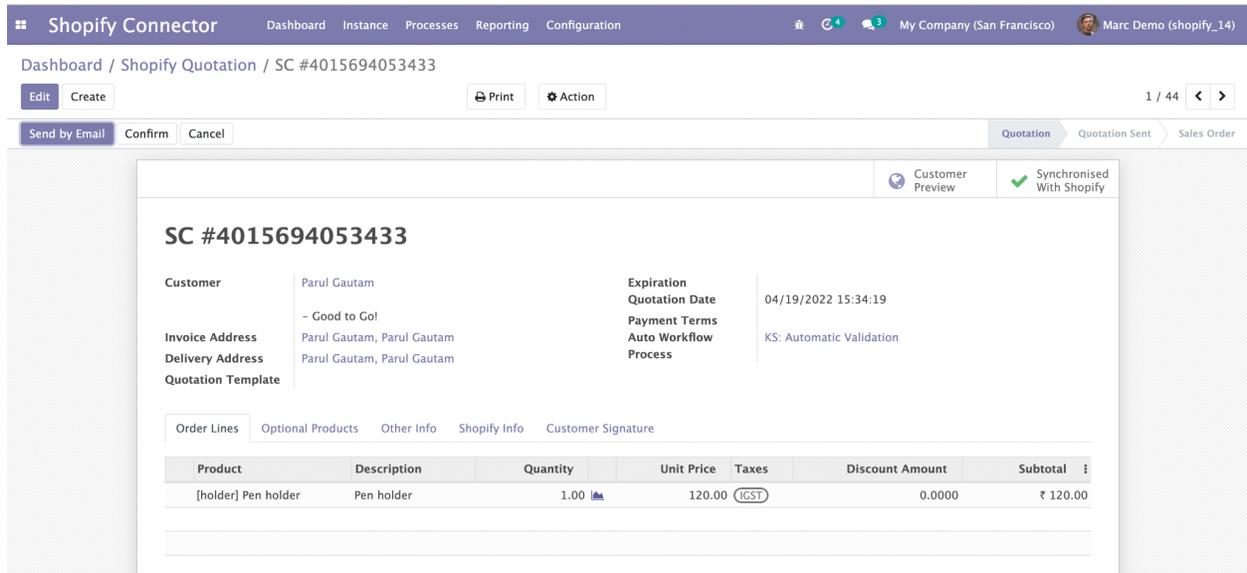


Here find your order and open it.

Number	Order Date	Customer	Salesperson	Instance	Shopify Id	Next Activity	Company	Total	Shopify Status	Sync Stat...	Invoice Status
SC #401569405...	04/19/2022	Parul Gautam	Mitchell Admin	K_dev	4015694053433		My Company (San...	₹ 141.60	Paid		Nothing to Invoice
SC #397383231...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3973832310841		My Company (San...	₹ 0.00	Paid		Fully Invoiced
SC #397483789...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974837895225		My Company (San...	₹ 23.60	Paid		Fully Invoiced
SC #397484376...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974843760697		My Company (San...	₹ 23.60	Pending		Fully Invoiced
SC #397484500...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974845005881		My Company (San...	₹ 23.60	Completed		Fully Invoiced
SC #397484605...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974846054457		My Company (San...	₹ 23.60	Pending		Fully Invoiced
SC #397850954...	04/19/2022	Pihu	Mitchell Admin	K_dev	3978509549625		My Company (San...	₹ 21.00	Completed		Fully Invoiced
SC #401566066...	04/19/2022	Parul Gautam	Mitchell Admin	K_dev	4015660662841		My Company (San...	₹ 141.60	Paid		Fully Invoiced
S00044	04/18/2022	Parul Gautam	Mitchell Admin	K_dev	4014983643193		My Company (San...	₹ 1.15	Completed		Fully Invoiced
SC #401497902...	04/18/2022	Parul Gautam	Mitchell Admin	K_dev	4014979022905		My Company (San...	₹ 1,416.00	Paid		Fully Invoiced
SC #401493340...	04/18/2022	Guest Customer	Mitchell Admin	K_dev	4014933409849		My Company (San...	₹ 1,416.00	Completed		Fully Invoiced
SC #397484500...	04/15/2022	Guest Customer	Joel Willis	K_dev	3974845005881		My Company (San...	\$ 23.60	Completed		Fully Invoiced
SC #401092303...	04/14/2022	A.D.Motors Spa	Mitchell Admin	K_dev	4010923032633		My Company (San...	₹ 120,000.00	Completed		To Invoice

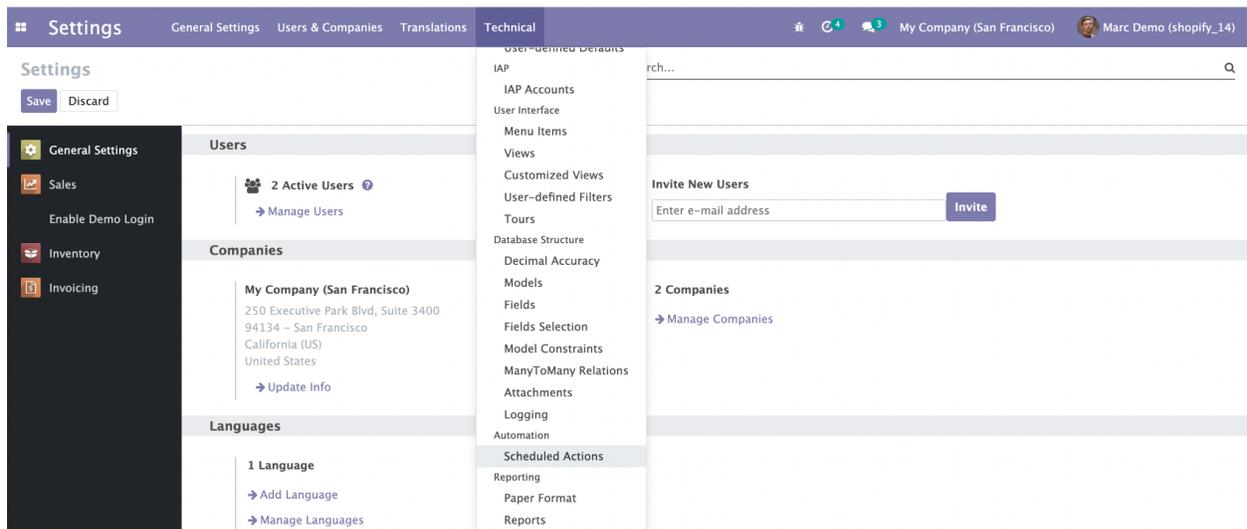
Once you open your order, you will see that it is in the Quotation state.

As the workflow is an automated process, after the set interval of time, your order will be changed to Sale Order.



If you want to manually run the scheduler.

Go to Settings>Technical>Scheduled Actions



Click on the Schedules Actions. Here, we are choosing the Ksolves Connector: Auto Invoice Workflow Job

Priority	Action Name	Model	Next Execution Date	Interval Number	Interval Unit	Number of Calls	Active
5	Account: Post draft entries with auto_post set to True up to t...	Journal Entry	04/20/2022 07:30:20	1	Days	-1	<input checked="" type="checkbox"/>
5	Base: Auto-vacuum internal data	Automatic Vacuum	04/19/2022 21:09:14	1	Days	-1	<input checked="" type="checkbox"/>
5	Digest Emails	Digest	04/19/2022 23:20:15	1	Days	-1	<input checked="" type="checkbox"/>
5	EDI : Perform web services operations	Electronic Document for an account.move	08/13/2021 21:20:27	1	Hours	-1	<input type="checkbox"/>
5	KS: Shopify Sales Report	Shopify Mail Report Configuration	04/19/2022 21:21:29	24	Hours	-1	<input checked="" type="checkbox"/>
5	Ksolves Connector: Auto Invoice Workflow Job	Sale WorkFlow Configuration	04/20/2022 01:20:37	10	Hours	-1	<input checked="" type="checkbox"/>
5	Mail: Email Queue Manager	Ksolves Connector: Auto Invoice Workflow Job ig Mails	04/19/2022 16:20:08	1	Hours	-1	<input checked="" type="checkbox"/>
5	Mail: Fetchmail Service	Incoming Mail Server	08/13/2021 21:20:10	5	Minutes	-1	<input type="checkbox"/>
1,000	Mail: Notify channel moderators	Message	04/19/2022 21:20:08	1	Days	-1	<input checked="" type="checkbox"/>
5	Notification: Delete Notifications older than 6 Month	Message Notifications	04/19/2022 21:20:08	1	Days	-1	<input checked="" type="checkbox"/>
5	Partner Autocomplete : Sync with remote DB	Partner Autocomplete Sync	04/19/2022 16:20:13	60	Minutes	-1	<input checked="" type="checkbox"/>
5	Post process payment transactions	Payment Transaction	04/19/2022 15:40:27	10	Minutes	-1	<input checked="" type="checkbox"/>
5	Procurement: run scheduler	Procurement Group	04/19/2022 21:20:25	1	Days	-1	<input checked="" type="checkbox"/>

Once it will open, you will find a 'Run Manually' button. Click on it and wait for a few nanoseconds (else the scheduler would have executed on the date-time set on the **next execution**)

Scheduled Actions / Ksolves Connector: Auto Invoice Workflow Job

6 / 58

Button : Run Manually

- Object: ir.cron
- Modifiers: [{"invisible": ["state", "l=", "code"]}]]
- Button Type: object
- Method: method\_direct\_trigger

**Ksolves Connector: Auto Invoice Workflow Job**

Model: Sale WorkFlow Configuration

Scheduler User: Odoobot

Execute Every: 10Hours

Active:

Next Execution Date: 04/20/2022 01:20:37

Number of Calls: -1

Priority: 5

Repeat Missed:

Python Code: Security Help

```
1 .ks_auto_process_sale_order()
```

Go to the Dashboards>Orders>Your Order. Now you will see that your order status has been changed from Quotation to Sale Order.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Shopify Quotation / SC #4015694053433

Edit Create Print Action 1 / 1 < >

Send by Email Cancel Quotation Quotation Sent Sales Order

Customer Preview 1 Delivery 1 Invoices Synchronised With Shopify

**SC #4015694053433**

Customer: Parul Gautam  
- Good to Go!  
Invoice Address: Parul Gautam, Parul Gautam  
Delivery Address: Parul Gautam, Parul Gautam  
Quotation Template

Order Date: 04/19/2022 15:34:19  
Payment Terms: KS: Automatic Validation  
Auto Workflow Process

Order Lines Other Info Shopify Info Customer Signature

Product	Description	Quantity	Delivered	Invoiced	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	1.00	1.00	120.00	IGST	0.0000	₹ 120.00

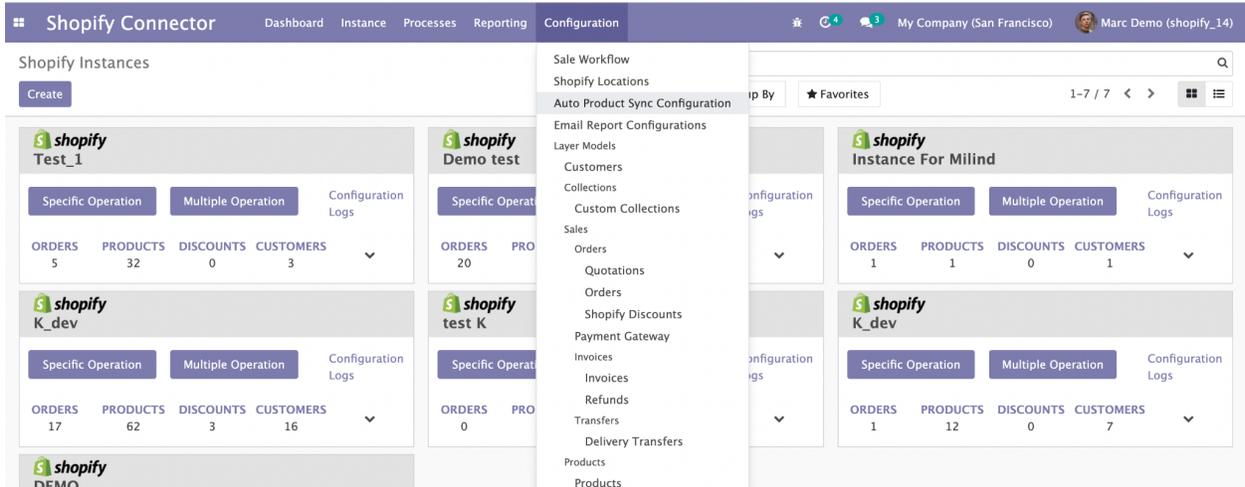
This is how the Sale Auto workflow functions are used to validate the Order status in Odoo.

## 8. Auto Product Sync

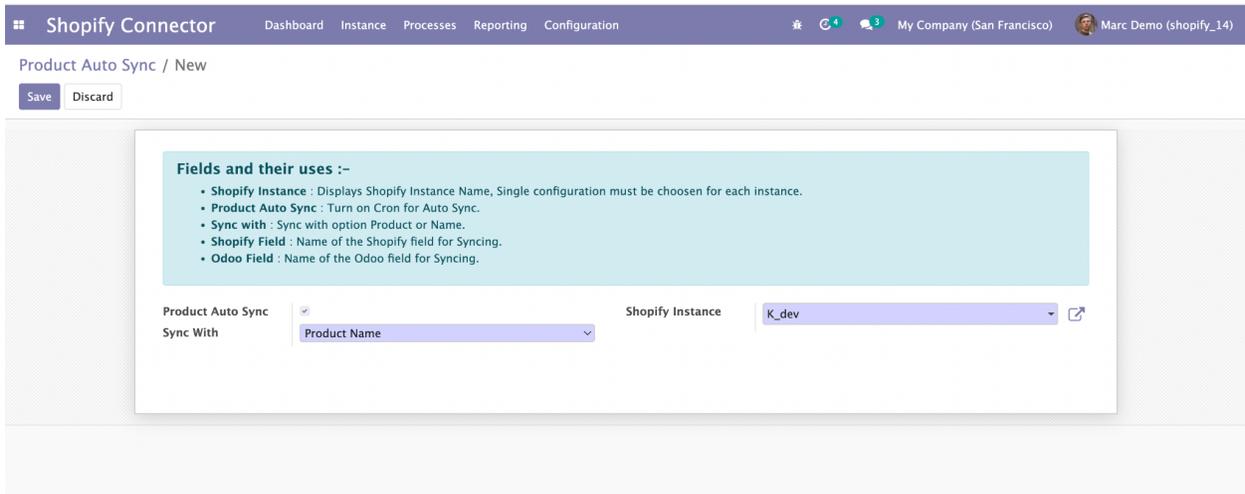
Many times two products with the same name might be present at the Odoo and Shopify ends but they aren't associated with one another (as neither of them will be imported or exported). In such a case, import products from Shopify would duplicate the product on Odoo's end.

To remove the duplicacy of two similar products, Auto product Sync Configuration is used.

Go to Shopify Connector>Configurations>Auto Product Sync Configuration



Click on the Create button, a window will appear as given below:



Select the Product Auto Sync option, without this we cannot syn the two similar products.

Select the 'Instance Id' and 'Sync With' options (which provides flexibility to map products by name/SKU/barcode), and click on the Save button.

You can create more than 1 Auto Sync configuration but for different instances. One instance cannot have more than one such configuration.

Once you will save the configuration, you will find the layer model containing all the Auto Sync configurations that you have created.

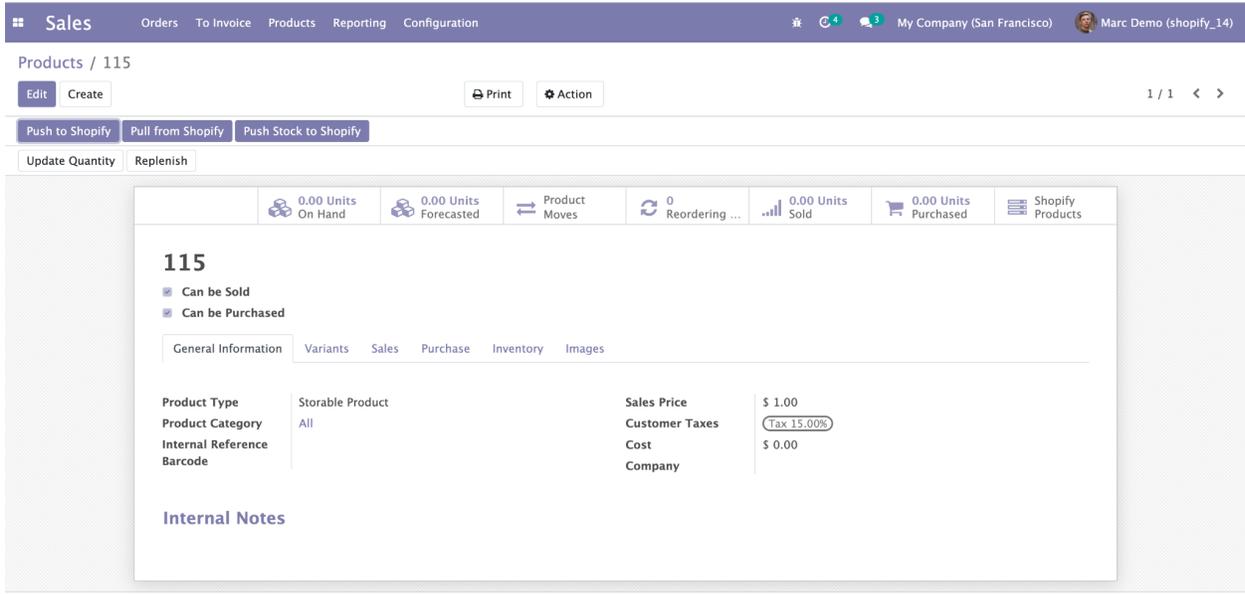
Shopify Connector		
Product Auto Sync		
Name	Shopify Instance	Product Auto Sync
<input type="checkbox"/> K_dev Auto Sync with Name	K_dev	<input checked="" type="checkbox"/>
<input type="checkbox"/> test K Auto Sync with Name	test K	<input checked="" type="checkbox"/>

Now create a new product in Shopify and Odoo with a similar name. Refer to section (4.1 a,c)

Newly created product in Shopify: Refer to section (4.1 a)

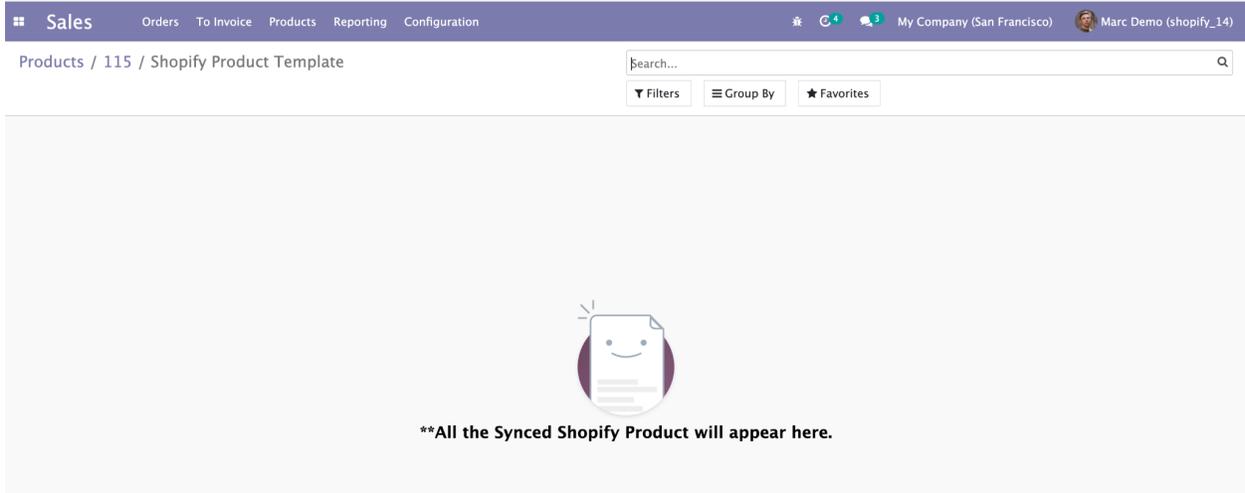
The screenshot shows the Shopify admin interface for a store named 'ksolves-dev-store...'. The user is Sanju Bharti. The main content area displays a newly created product with ID 115, which is active. The product title is '115' and the description is 'All OK!'. The product status is set to 'Active'. The interface includes a sidebar with navigation options like Home, Orders, Products, Customers, Analytics, Marketing, Discounts, and Apps. A notification banner at the top says 'Added 115'.

Newly created product in Odoo: Refer to (4.2 a)



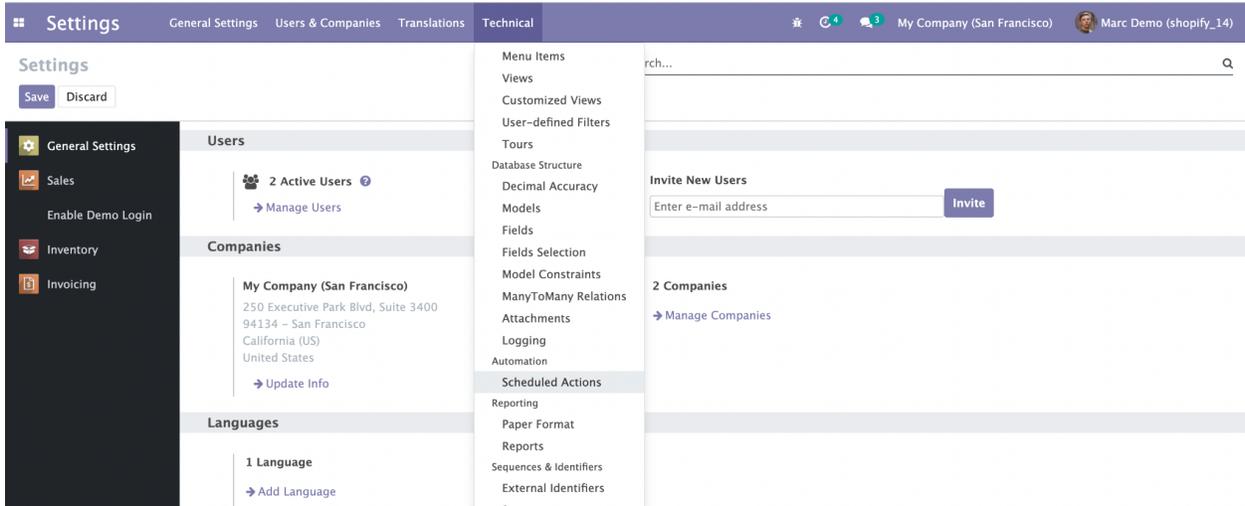
If you will click on the Shopify Product at the top right corner, a window will appear as given below:

It shows that the product has not yet been synced.



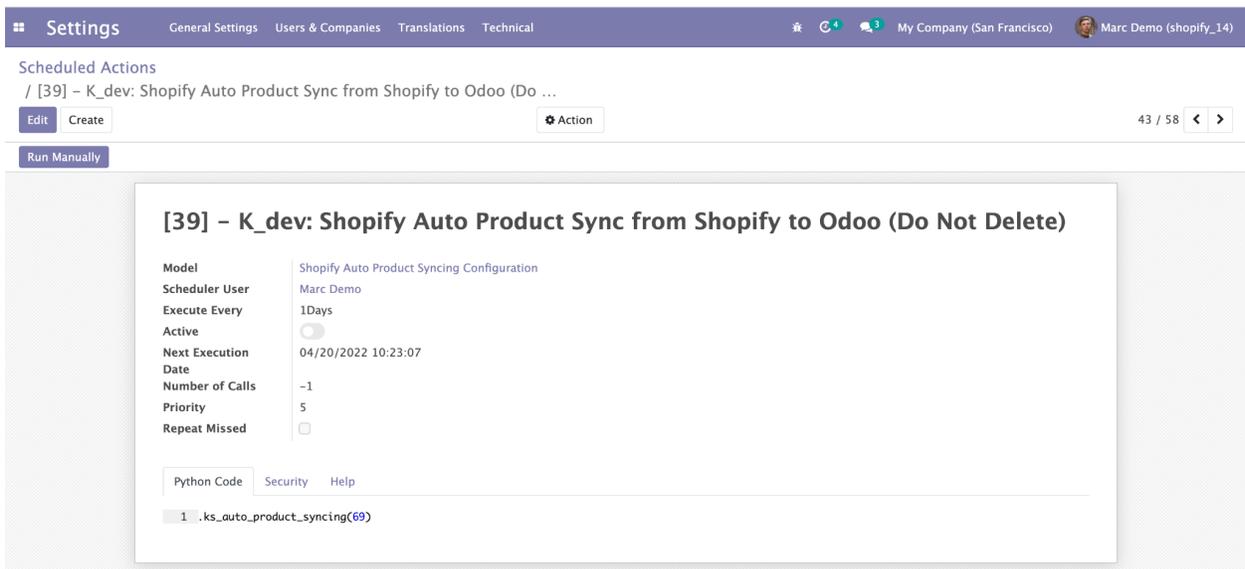
Here's a manual process to execute the scheduled Action:

Go to Settings>Technical> Scheduled Actions



Select the Scheduled Action and open it.

You will see that your scheduled action is deactivated.



Either activate the Scheduled Action and wait till the next execution Date for the cron to execute or click on the 'Run Manually' button.

The screenshot shows the Odoo Settings interface. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. The user is logged in as 'Marc Demo (shopify\_14)'. The main content area is titled 'Scheduled Actions' and shows a list of actions. The selected action is '[39] - K\_dev: Shopify Auto Product Sync from Shopify to Odoo (Do Not Delete)'. A modal window displays the configuration for this action:

- Model:** Shopify Auto Product Syncing Configuration
- Scheduler User:** Marc Demo
- Execute Every:** 1Days
- Active:**
- Next Execution Date:** 04/20/2022 10:23:07
- Number of Calls:** -1
- Priority:** 5
- Repeat Missed:**

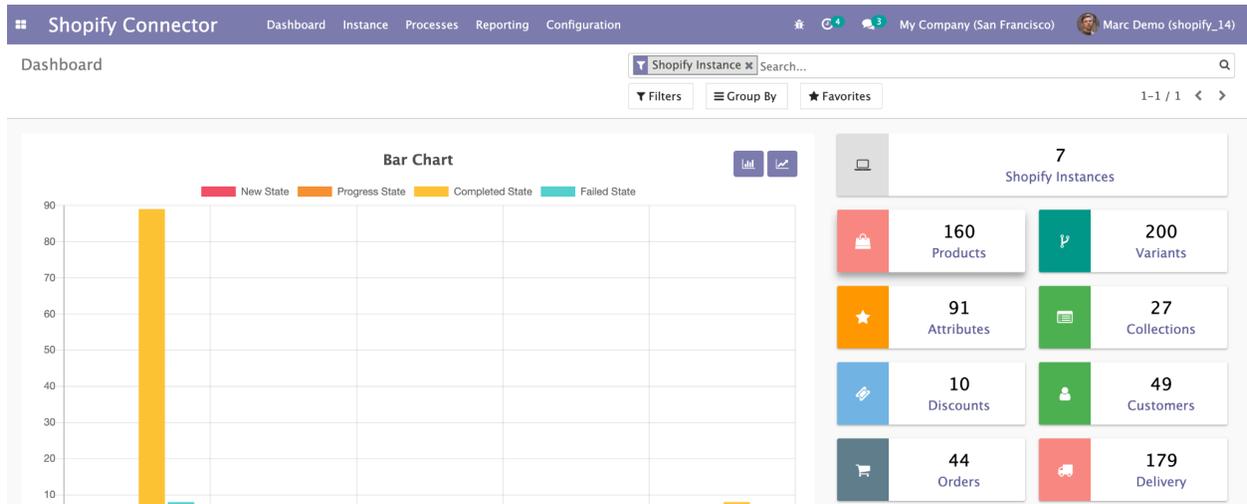
Below the configuration, there are tabs for 'Python Code', 'Security', and 'Help'. The 'Python Code' tab is active, showing a single line of code: `1 .ks_auto_product_syncing(69)`.

Now go to your product and click on the Shopify Product as mentioned before. Here you will find that a layer model is created that shows that the product is now syncing.

The screenshot shows the Odoo Sales dashboard. The top navigation bar includes 'Sales', 'Orders', 'To Invoice', 'Products', 'Reporting', and 'Configuration'. The user is logged in as 'Marc Demo (shopify\_14)'. The main content area is titled 'Products / 115 / Shopify Product Template'. A search bar and navigation controls are visible. Below, a table displays the product templates:

Shopify Product ID	Odoo Product Template	Instance	Date Created	Date Updated
6732181569593	115	K_dev	04/19/2022 19:30:25	04/19/2022 19:30:26

To verify, you can go to the dashboard and click on Products.



Here you will find that your product is synced under the given instance.

The interface shows the configuration for a product template with ID 115. It includes a list of fields and their uses, and a table of configuration details.

**Fields and their uses :-**

- **Odoo Product Template** : Displays Odoo Linked Product Template Name
- **Shopify Product Type** : Displays Shopify Product Type
- **Shopify Product ID** : Displays Shopify ID
- **Instance** : Displays Shopify Instance Name
- **Date Created** : Displays date of creation
- **Date Updated** : Displays date of updation
- **Tags** : Displays Shopify Tags
- **Shopify Regular Price** : Displays Shopify Regular Price
- **Shopify Sale Price** : Displays Shopify Sale Price

Odoo Product Template	115	Shopify Product ID	6732181569593
Shopify Product Type	Simple Product	Collection Ids	
Product Type		Shopify Product Active Status	<input checked="" type="checkbox"/>
Tags		Instance	K_dev
Vendor	ksolves-dev-store-001	Inventory ID	42191929802809
Inventory Policy	Deny	Date Created	04/19/2022 19:30:25
		Date Updated	04/19/2022 19:30:26

Go to Processes>Queue Jobs

The list will show that the duplicacy of data is prevented and a single product with a similar name is synced between Odoo and Shopify.

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
115	6732181569593	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
114	6732163973177	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
113	6732153389113	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
112	6732135465017	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
112	6732135465017	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
112	6732135465017	Product Template	Shopify to Shopify Layer	Import	K_dev	Done
112	6732135465017	Product Template	Shopify to Shopify Layer	Import	K_dev	Done

## 9. Report

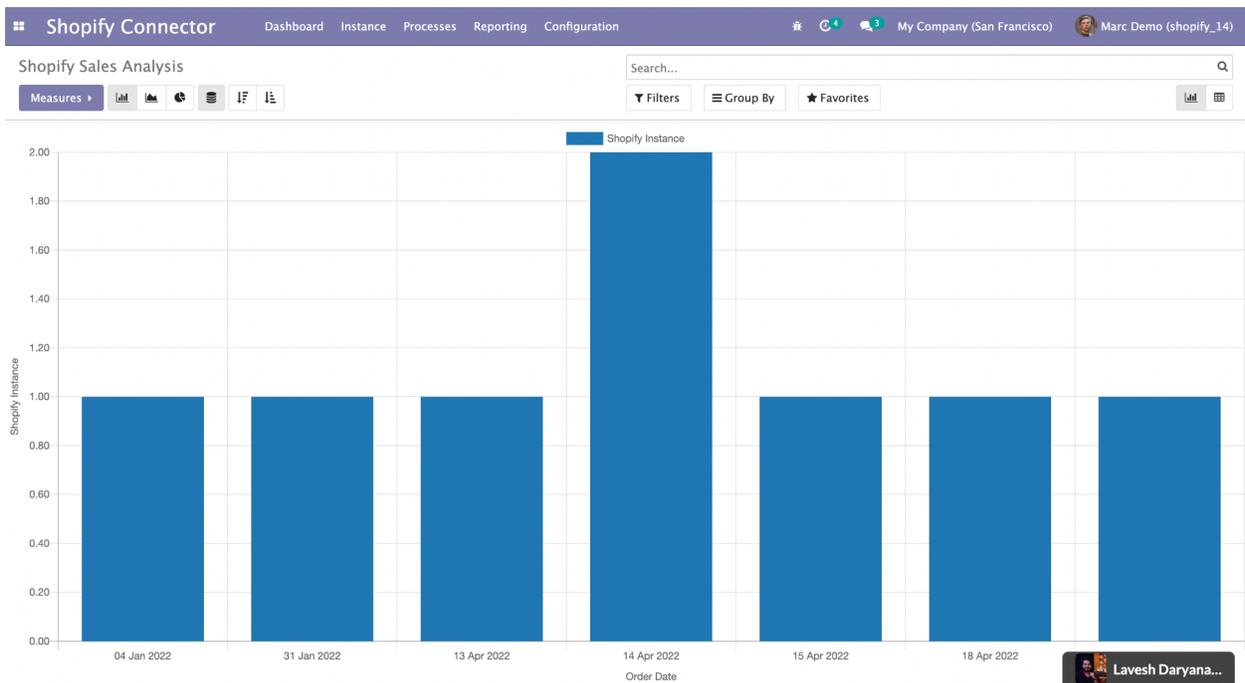
Regular monitoring of your sales is essential for any business. Make use of the specialized Sale Report Analysis in Odoo to learn how your business is doing.

### 9.1 Sale Reporting

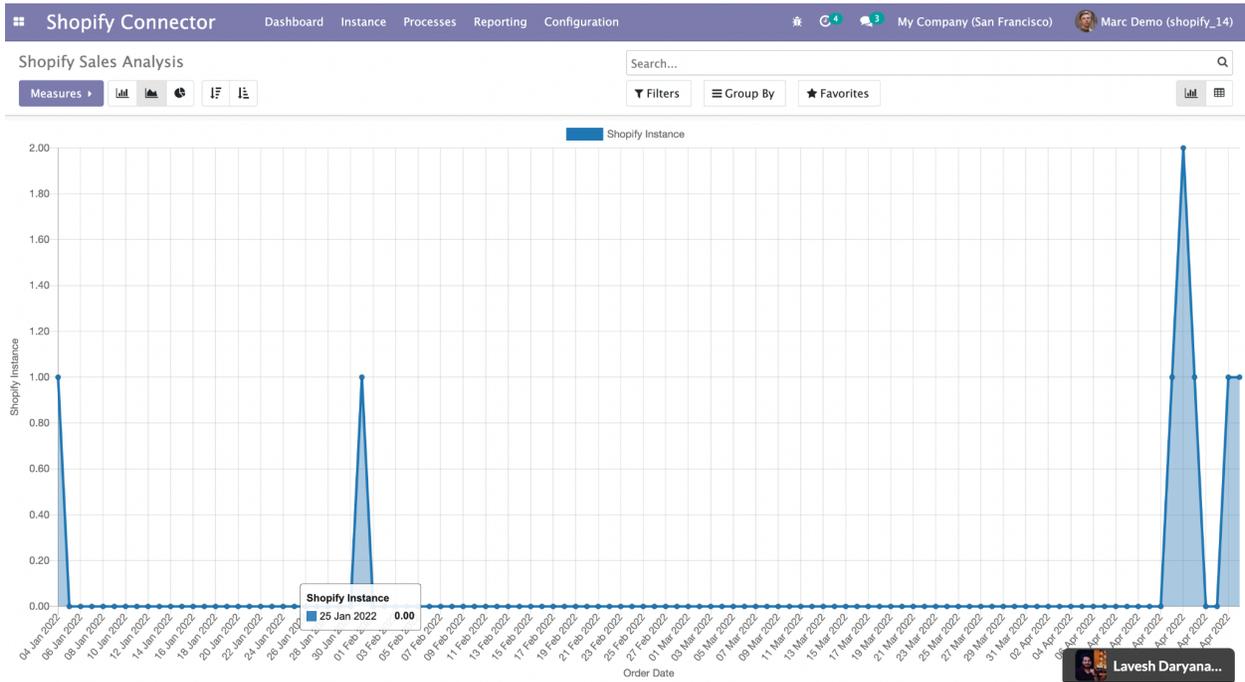
Go to Shopify Connector>Reporting>Sale

Here you will find the graphical representation of your sale. You can convert it into Bar Chart, Line Chart, and Pie Chart. Or you can get a list view of your reports.

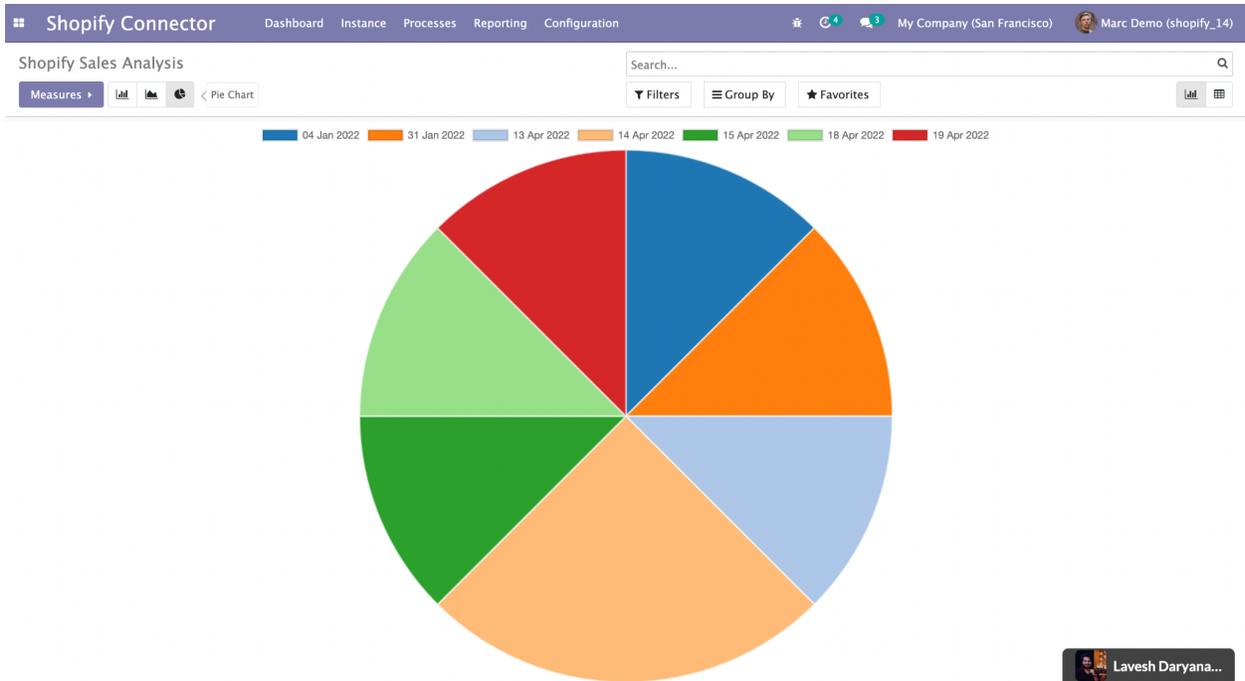
#### a. Bar Chart



## b. Line Chart



## c. Pie Chart



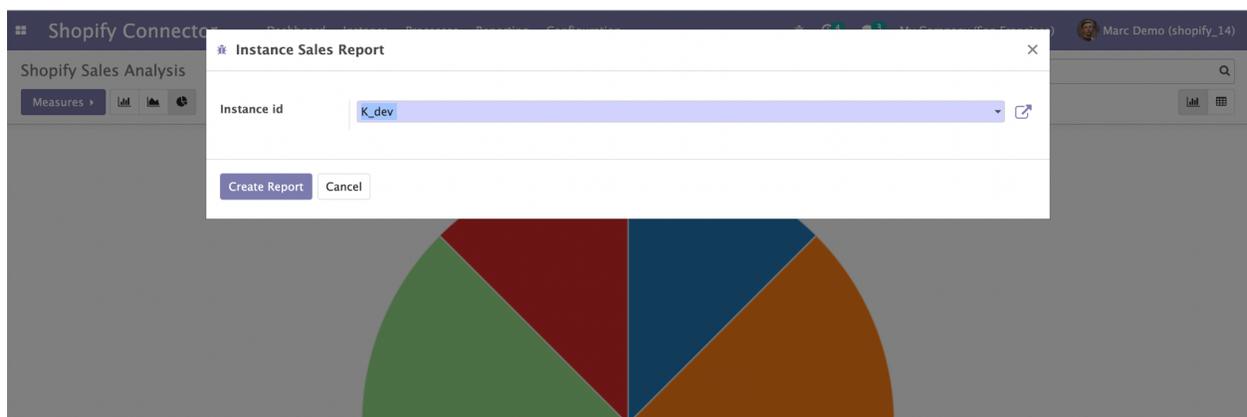
## d. Pivot View

- Total		+ Europe		+ America			
	Product Variant	Untaxed Total	Product Variant	Untaxed Total	Product Variant	Untaxed Total	
- Total		13	6,119.05	10	688.65	23	6,807.70
- January 2022		7	534.33	9	588.65	16	1,122.98
+ GauravK singh				1	2.29	1	2.29
+ Guest Customer		1	14.28	7	586.33	8	600.61
+ Jiya Bharti		1	0.64			1	0.64
+ Jiya Bharti (2)		1	0.64			1	0.64
+ K a s h i s h Kashish		3	4.84			3	4.84
+ Piya Chauhan		1	1.53			1	1.53
+ ROCKY				1	0.03	1	0.03
+ S i y a		2	17.34			2	17.34
+ d e h r a m		3	258.62			3	258.62
+ k a s h i s h		2	1.78			2	1.78
+ kashish ahuja		1	3.19			1	3.19
+ meenal ahuja		2	27.53			2	27.53
+ sakshi dhoni		2	203.94			2	203.94
+ sanju bharti		1	0.00			1	0.00
- April 2022		6	5,584.72	1	100.00	7	5,684.72
+ A.D.Motors Spa		1	4,259.13			1	4,259.13

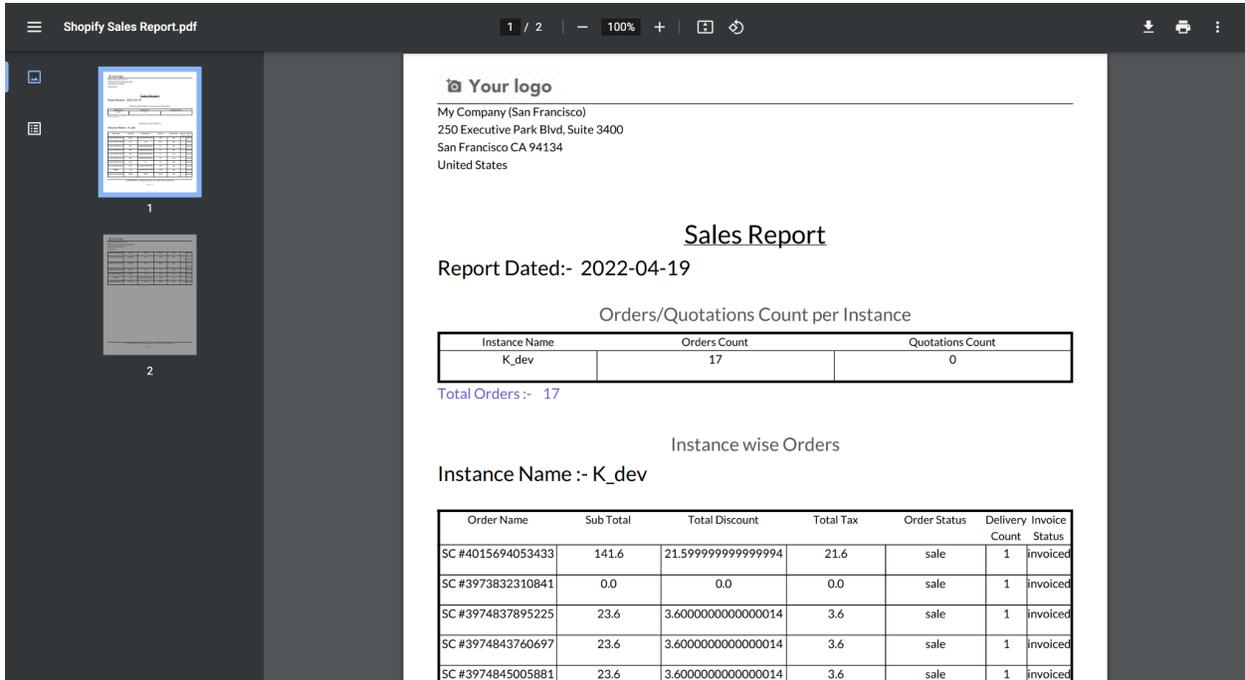
## 9.2 Instance Wise Report

Go to Shopify Connector>Reporting>Instance Sales Report, and a wizard will open.

Chose the instance for which you need to create the sales report.

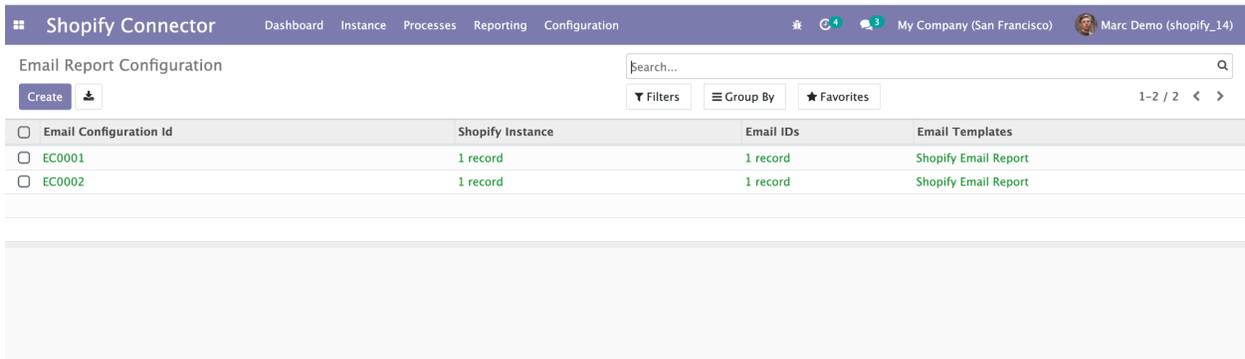


Click on the 'Create Report' button, and you will get a sales report in PDF format.



### 9.3 Email Report Configuration

If you want to email the sale report, you can by visiting the Configuration>Email Configuration



Create your email configuration and fill in the required details as shown below:

**Shopify Connector** | Dashboard | Instance | Processes | Reporting | Configuration | My Company (San Francisco) | Marc Demo (shopify\_14)

Email Report Configuration / EC0001

Buttons: Edit, Create, Print, Action, Send Email

**Fields and their uses :-**

- **Shopify Instance** : Displays Shopify Name
- **Email IDs** - Displays list of email ID's
- **Email Templates** : Displays Email Template name
- **Email Subject** : Display Subject of the Email
- **Email Body** : Display Body of the Email

**Basic Configuration**

**EC0001**

Shopify Instance: Demo test

Email IDs: Demoserver@gmail.com

Email Templates: Shopify Email Report

Email Subject & Body

Email Subject: aaaaaaaaa

Email Body:

For manual triggering of the Mail cron: Go to Settings>Technical>Scheduled Actions. Select the email scheduled action related to your Shopify.

**Settings** | General Settings | Users & Companies | Translations | Technical | My Company (San Francisco) | Marc Demo (shopify\_14)

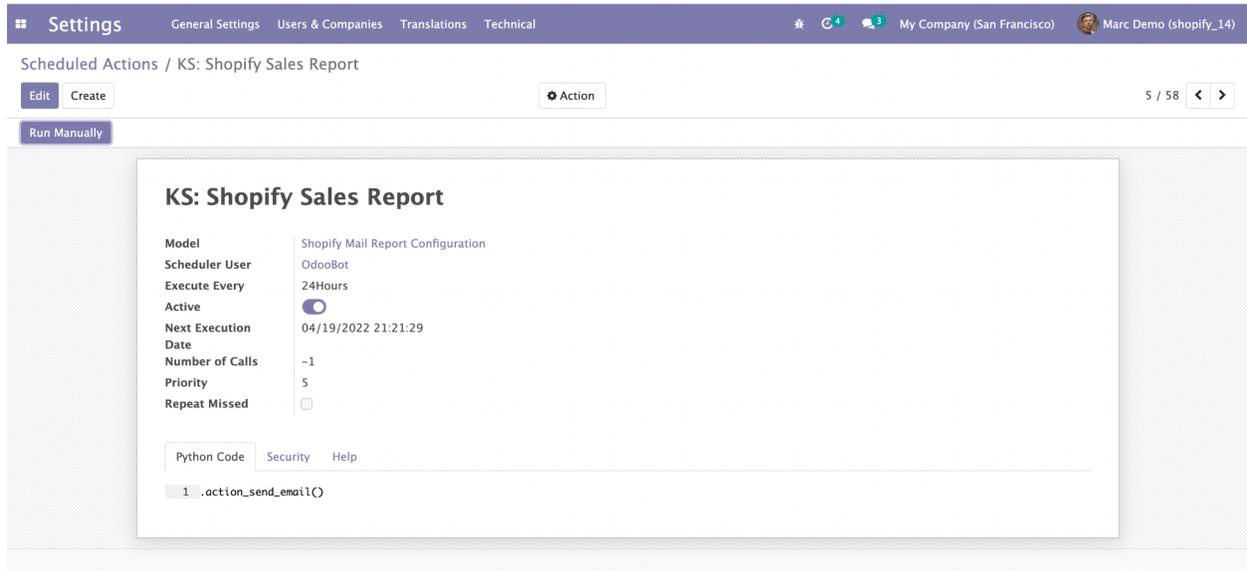
Scheduled Actions

Buttons: Create, Filters, Group By, Favorites

Priority	Action Name	Model	Next Execution Date	Interval Number	Interval Unit	Number of Calls	Active
5	Account; Post draft entries with auto_post set to True up to t...	Journal Entry	04/20/2022 07:30:20	1	Days	-1	<input checked="" type="checkbox"/>
5	Base: Auto-vacuum internal data	Automatic Vacuum	04/19/2022 21:09:14	1	Days	-1	<input checked="" type="checkbox"/>
5	Digest Emails	Digest	04/19/2022 23:20:15	1	Days	-1	<input checked="" type="checkbox"/>
5	EDI : Perform web services operations	Electronic Document for an account.move	08/13/2021 21:20:27	1	Hours	-1	<input type="checkbox"/>
5	KS: Shopify Sales Report	Shopify Mail Report Configuration	04/19/2022 21:21:29	24	Hours	-1	<input checked="" type="checkbox"/>
5	Ksolves Connector: Auto Invoice V	Sale WorkFlow Configuration	04/20/2022 01:20:37	10	Hours	-1	<input checked="" type="checkbox"/>
5	Mail: Email Queue Manager	Outgoing Mails	04/19/2022 20:20:08	1	Hours	-1	<input checked="" type="checkbox"/>
5	Mail: Fetchmail Service	Incoming Mail Server	08/13/2021 21:20:10	5	Minutes	-1	<input type="checkbox"/>
1,000	Mail: Notify channel moderators	Message	04/19/2022 21:20:08	1	Days	-1	<input checked="" type="checkbox"/>
5	Notification: Delete Notifications older than 6 Month	Message Notifications	04/19/2022 21:20:08	1	Days	-1	<input checked="" type="checkbox"/>
5	Partner Autocomplete : Sync with remote DB	Partner Autocomplete Sync	04/19/2022 20:20:13	60	Minutes	-1	<input checked="" type="checkbox"/>
5	Post process payment transactions	Payment Transaction	04/19/2022 19:30:27	10	Minutes	-1	<input checked="" type="checkbox"/>

Once you find your scheduled email action, open it, there you will see the time it is set to send the email automatically.

If you wish to email the sale report manually, click on the Run Manually button and send the report to the concerned person.



The screenshot shows the Odoo Settings interface for a scheduled action named 'KS: Shopify Sales Report'. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. The user is logged in as 'Marc Demo (shopify\_14)'. The page title is 'Scheduled Actions / KS: Shopify Sales Report'. There are buttons for 'Edit', 'Create', and 'Action'. A 'Run Manually' button is highlighted. The main content area displays the configuration for the 'KS: Shopify Sales Report' action:

Model	Shopify Mail Report Configuration
Scheduler User	Odoobot
Execute Every	24Hours
Active	<input checked="" type="checkbox"/>
Next Execution Date	04/19/2022 21:21:29
Number of Calls	-1
Priority	5
Repeat Missed	<input type="checkbox"/>

Below the configuration table, there are tabs for 'Python Code', 'Security', and 'Help'. The 'Python Code' tab is active, showing a single line of code: `1 .action_send_email()`.

This way you can keep your business going by keeping a record of sales.

## 10. Stock Refund

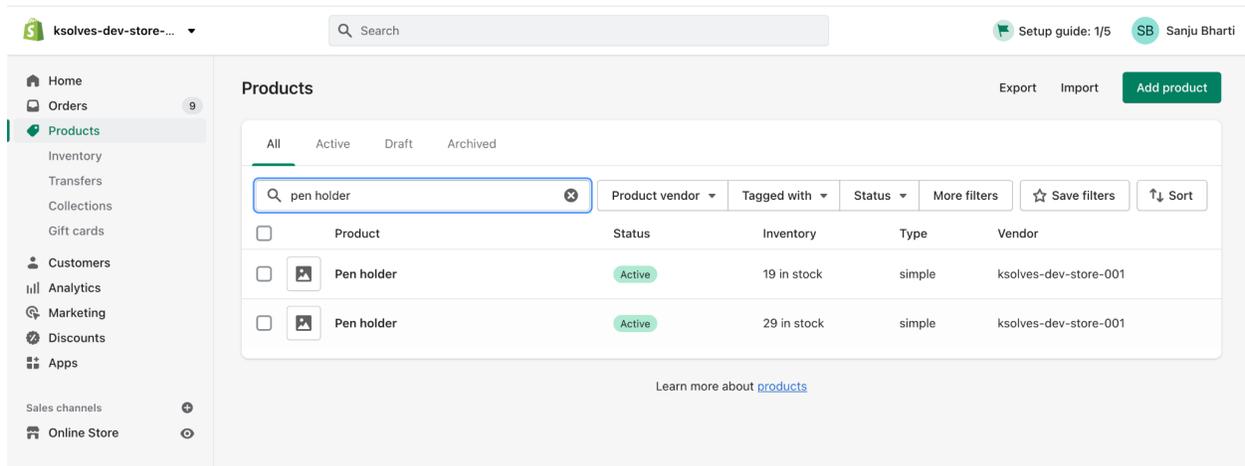
Many times it happens that a customer returns order and a refund has to be generated for the same. In such scenarios, eCommerce store owners need to update the stock in the inventory after the product is refunded.

With the Odoo Shopify connector, a user can update the stock in Shopify from Odoo's ends easily.

**Note:** Before 'Stock Refund', it is necessary to sync the Product and Product Stock between Odoo and Shopify. For Product Stock and Order Import refer to sections 4.1 b and 4.1 c respectively.

Go to Shopify end and find a product you wish to place an order for.

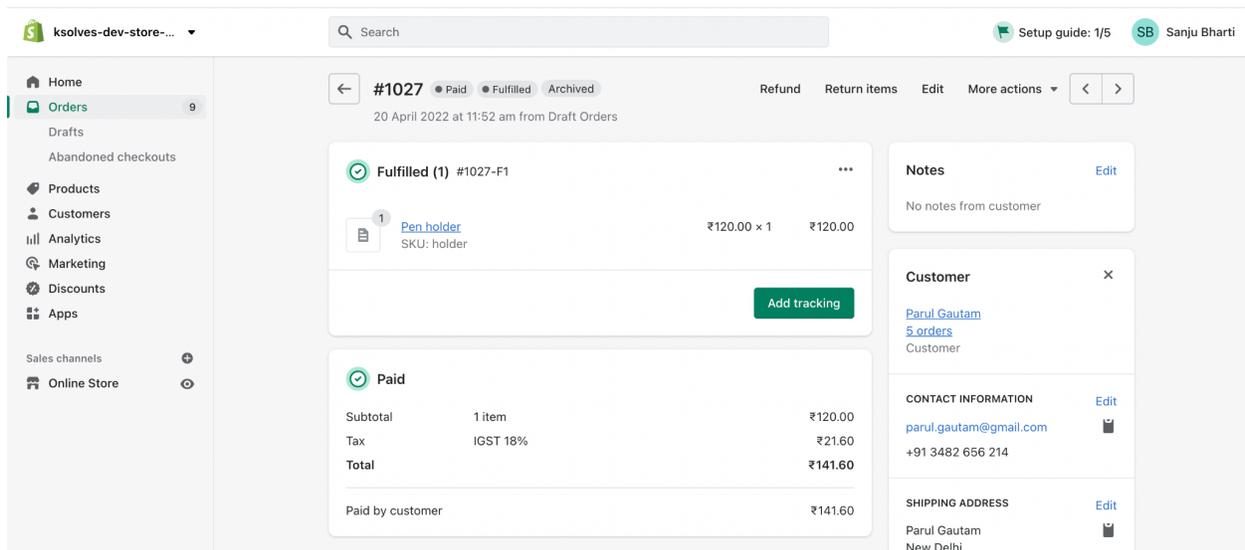
For example, we have taken a Pen holder with 29 quantities in stock.



The screenshot shows the Shopify admin interface for a store named 'ksolves-dev-store...'. The 'Products' section is active, displaying a search for 'pen holder'. The search results table is as follows:

Product	Status	Inventory	Type	Vendor
<input type="checkbox"/> Pen holder	Active	19 in stock	simple	ksolves-dev-store-001
<input type="checkbox"/> Pen holder	Active	29 in stock	simple	ksolves-dev-store-001

Create an order (section 4.1 c) and fulfill the order at the Shopify end.



The screenshot shows the Shopify admin interface for an order with ID #1027. The order is marked as 'Paid' and 'Fulfilled'. The order details are as follows:

**Order #1027** (Paid, Fulfilled, Archived)  
20 April 2022 at 11:52 am from Draft Orders

**Fulfilled (1) #1027-F1**

1 Pen holder SKU: holder	₹120.00 x 1	₹120.00
-----------------------------	-------------	---------

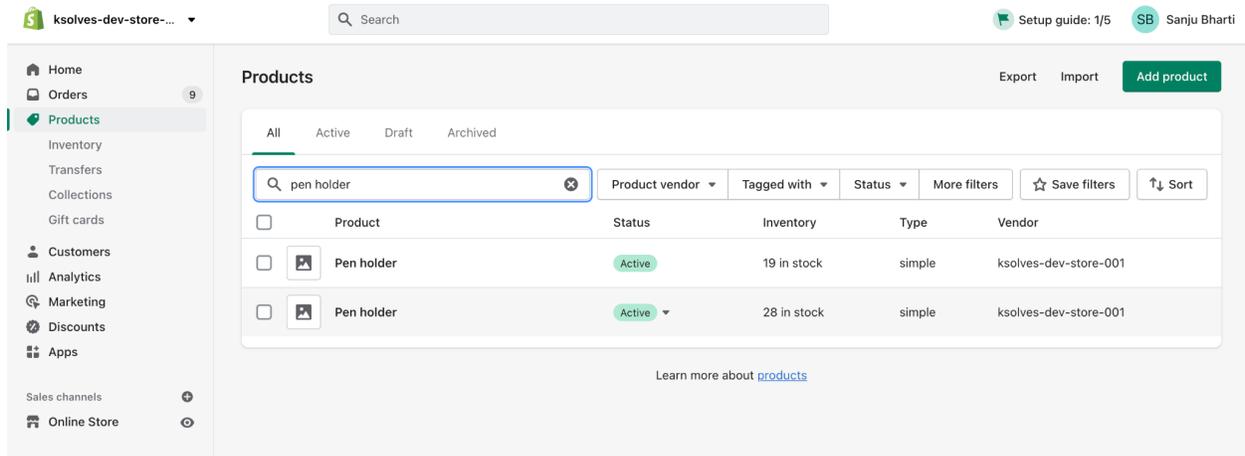
**Paid**

Subtotal	1 item	₹120.00
Tax	IGST 18%	₹21.60
<b>Total</b>		<b>₹141.60</b>

Paid by customer ₹141.60

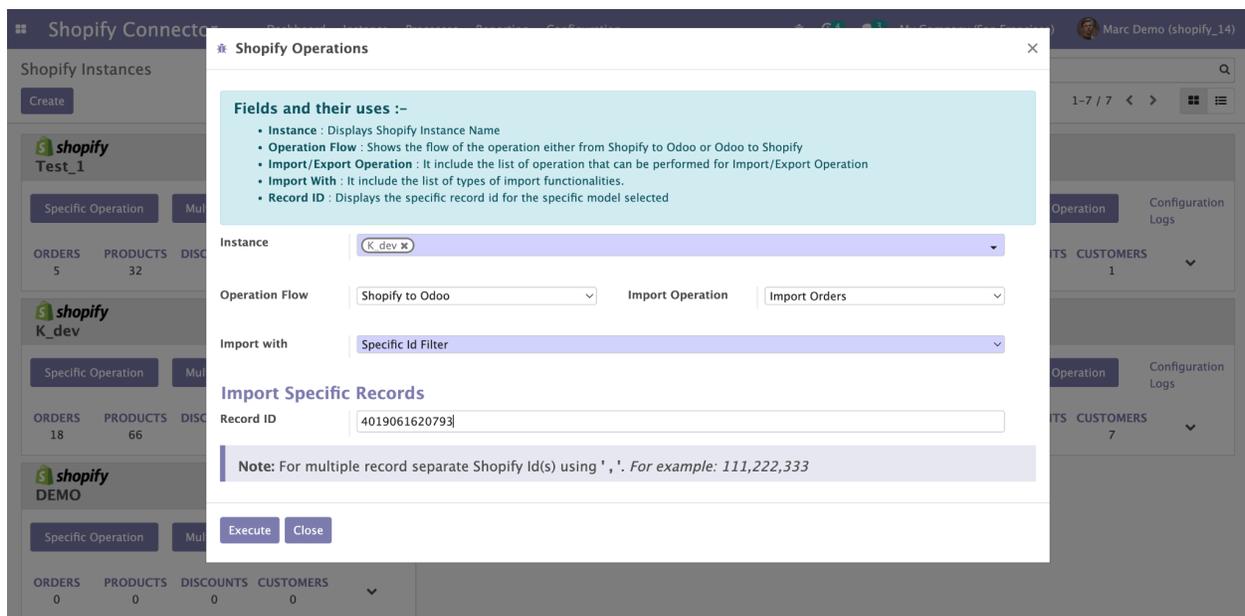
**Customer:** Parul Gautam (5 orders)  
**CONTACT INFORMATION:** parul.gautam@gmail.com, +91 3482 656 214  
**SHIPPING ADDRESS:** Parul Gautam, New Delhi

After fulfilling the order, check the product stock again. Now you will see here that our product stock has changed from 29 to 28.

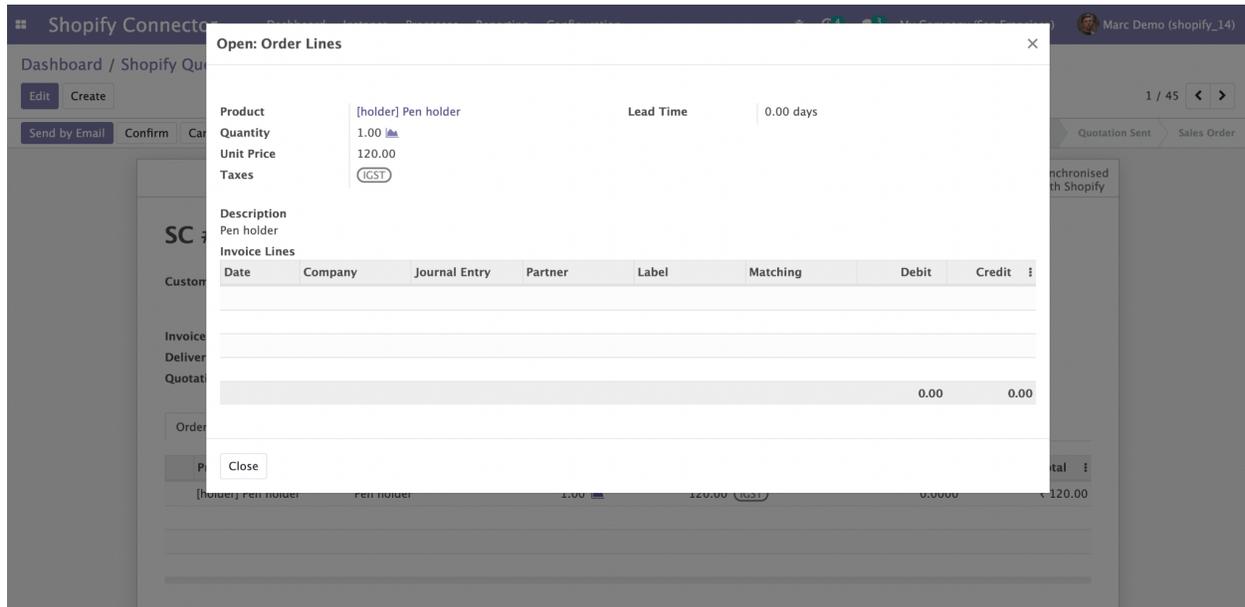


## a. Import Order

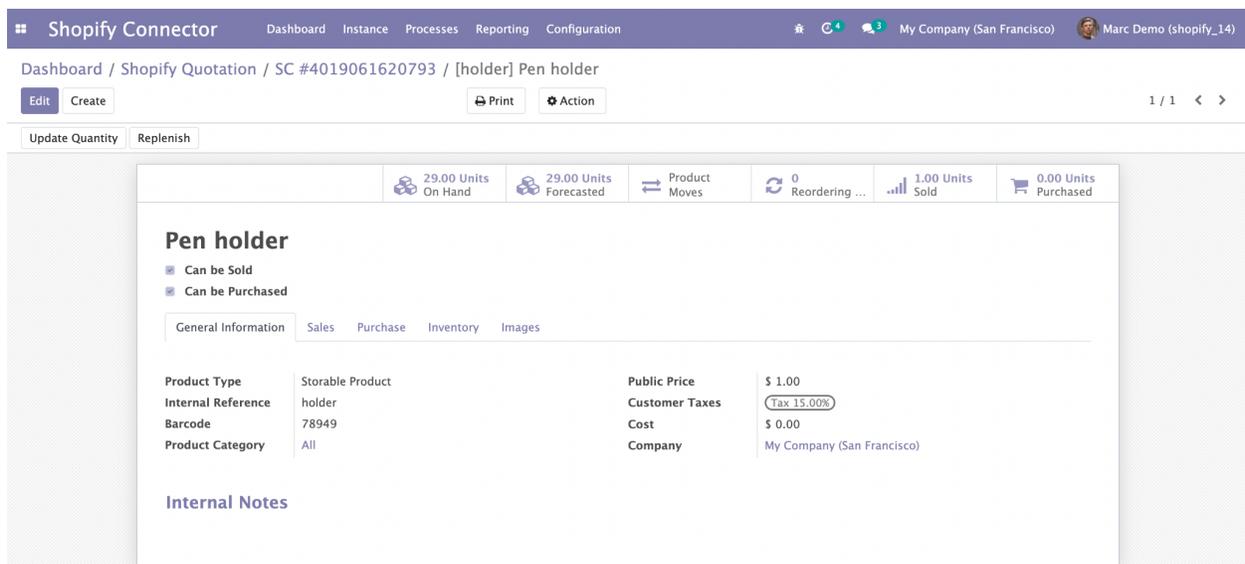
Go to Shopify Instance and import the newly created order.



Now go to Dashboard>Orders>Open the newly created order and click on the order line. You will get a window containing the order details as given below:

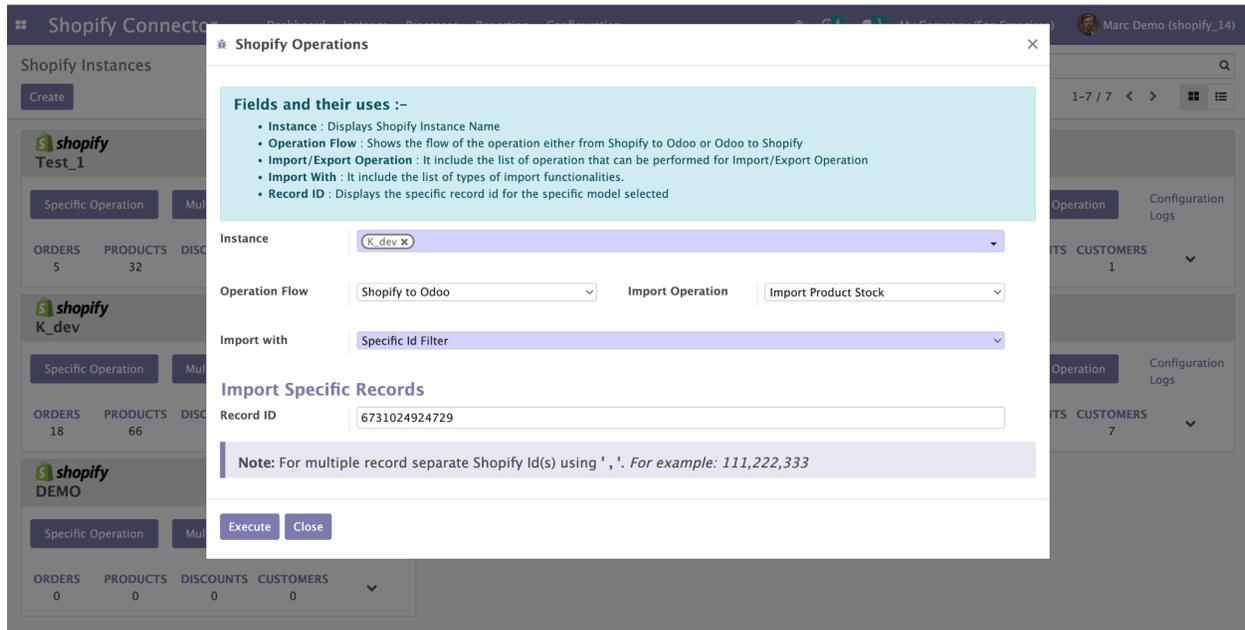


If you click on the [holder]Pen holder you will see a page like this. Here we can see the product stock has not been updated. The pen holder quantity is still 29 at the Odoo end.



## b. Update Stock

To update the product stock, import the product stock from Shopify to Odoo. Refer to section (4.1 b).



You can verify your stock import from Processes > Queue jobs

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

All Queue Jobs Search...

Filters Group By Favorites 1-12 / 12

Name	Shopify ID	Domain	Operation Flow	Operation Type	Shopify Instance	State
<input type="checkbox"/> Inventory Adjustment	0	stock	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Sale Order #4019061620793	4019061620793	Sale Order	Shopify To Odoo	Import	K_dev	Failed
<input type="checkbox"/> Parul	5388232949817	Customer	Shopify to Shopify Layer	Import	K_dev	Done
<input type="checkbox"/> Inventory Adjustment	0	stock	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Done
<input type="checkbox"/> Sale Order #4015694053433	4015694053433	Sale Order	Shopify To Odoo	Import	K_dev	Failed

### c. Validate Inventory

Now, go to Inventory>Operations>Inventory Adjustments, and click on your product inventory.

A window will appear as given below. As you can see the product inventory is in the progress state, to validate, click on the 'Validate Inventory' button.

The screenshot shows a web application interface for 'Inventory Adjustments'. The top navigation bar includes 'Inventory', 'Overview', 'Operations', 'Products', 'Reporting', and 'Configuration'. The user is logged in as 'Marc Demo (shopify\_14)' for 'My Company (San Francisco)'. The breadcrumb trail is 'Inventory Adjustments / product\_inventory\_2022-04-20 06:36:21'. Below the breadcrumb are buttons for 'Edit', 'Create', 'Print', and 'Action'. A progress indicator shows '1 / 41' with left and right arrows. The main form area has tabs: 'Continue Inventory', 'Validate Inventory' (selected), 'Print Count Sheet', and 'Cancel Inventory'. On the right, the status is 'Draft', 'In Progress', and 'Validated'. The form content includes: 'product\_inventory\_2022-04-20 06:36:21', 'Products' field with '(holder) Pen holder', 'Include Exhausted Products' checkbox (unchecked), 'Accounting Date Company' dropdown set to 'My Company (San Francisco)', and a 'Product Moves' button with a double-headed arrow icon.

Once you click on the 'Validate Inventory' button, the state will be changed to 'Validated'.

This screenshot is identical to the previous one, but the 'Validate Inventory' tab is now highlighted in blue, and the 'Validated' status in the top right corner is also highlighted in blue, indicating the form's state has been updated.

Go back to the Sales>Products>Products>(search your product), and you will see that the product stock has been updated and changed from 29 to 28.

The screenshot displays the 'Inventory' management interface for a product named 'Pen holder'. The top navigation bar includes 'Overview', 'Operations', 'Products', 'Reporting', and 'Configuration'. The user is logged in as 'Marc Demo (shopify\_14)' for 'My Company (San Francisco)'. The product page shows '28.00 Units On Hand' and '28.00 Units Forecasted'. A 'Product Moves' section shows '0 Reordering ...' and '1.00 Units Sold'. A 'Shopify Products' section shows '0.00 Units Purchased'. The product details include 'Can be Sold' and 'Can be Purchased' checkboxes, both checked. The 'General Information' tab is active, showing 'Product Type: Storable Product', 'Product Category: All', 'Internal Reference: holder', 'Barcode: 78949', 'Sales Price: \$ 1.00', 'Customer Taxes: Tax 15.00%', 'Cost: \$ 0.00', and 'Company: My Company (San Francisco)'. There is also an 'Internal Notes' section.

Go to Dashboard>Orders>(click on your order).

The Order is in the Quotation state. Click on the 'Confirm' button and complete the Sale Order.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793

Edit Create Print Action 1 / 45

Send by Email Confirm Cancel Quotation Quotation Sent Sales Order

Customer Preview Synchronised With Shopify

### SC #4019061620793

Customer: Parul Gautam  
- Good to Go!

Invoice Address: Parul Gautam, Parul Gautam  
Delivery Address: Parul Gautam, Parul Gautam  
Quotation Template

Expiration Date: 04/20/2022 11:52:38  
Payment Terms: KS: Automatic Validation  
Auto Workflow Process

Order Lines Optional Products Other Info Shopify Info Customer Signature

Product	Description	Quantity	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	120.00	IGST	0.0000	₹ 120.00

## d. Validate Delivery

To confirm the delivery, click on the '1 Delivery' tab.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793

Edit Create Print Action 1 / 45

Create Invoice Send by Email Cancel Quotation Quotation Sent Sales Order

Customer Preview 1 Delivery Synchronised With Shopify

### SC #4019061620793

Customer: Parul Gautam  
- Good to Go!

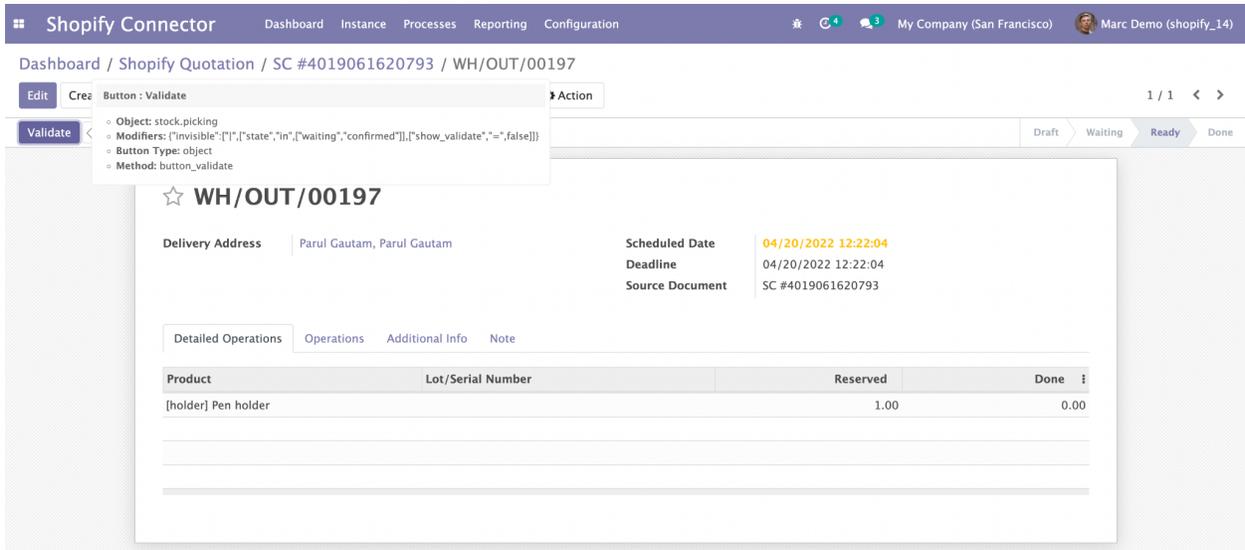
Invoice Address: Parul Gautam, Parul Gautam  
Delivery Address: Parul Gautam, Parul Gautam  
Quotation Template

Order Date: 04/20/2022 12:22:04  
Payment Terms: KS: Automatic Validation  
Auto Workflow Process

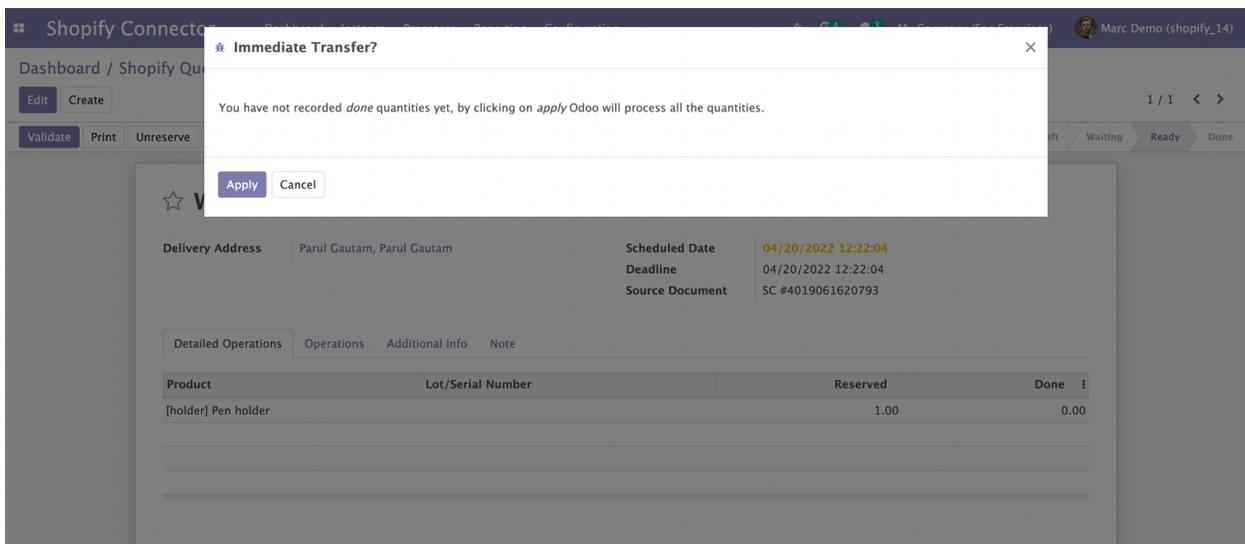
Order Lines Other Info Shopify Info Customer Signature

Product	Description	Quantity	Delivered	Invoiced	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	0.00	0.00	120.00	IGST	0.0000	₹ 120.00

Now confirm the delivery by clicking on the 'Validate' button.



When you click on the Validate button a dialog box will appear as shown below:  
Click on the apply button.



The order delivery state will change from Ready to Done state.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793 / WH/OUT/00197

Edit Create Print Action 1 / 1 < >

Print Return Unlock Draft Waiting Ready Done

\$ Valuation

☆ WH/OUT/00197

Delivery Address Parul Gautam, Parul Gautam

Scheduled Date 04/20/2022 12:22:04  
 Effective Date 04/20/2022 12:36:22  
 Source Document SC #4019061620793

Detailed Operations Operations Additional Info Note

Product	Lot/Serial Number	Reserved	Done
[holder] Pen holder		0.00	1.00

Go back to Dashboard>Order>(click on your order)

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation

Search... 1-45 / 45 < >

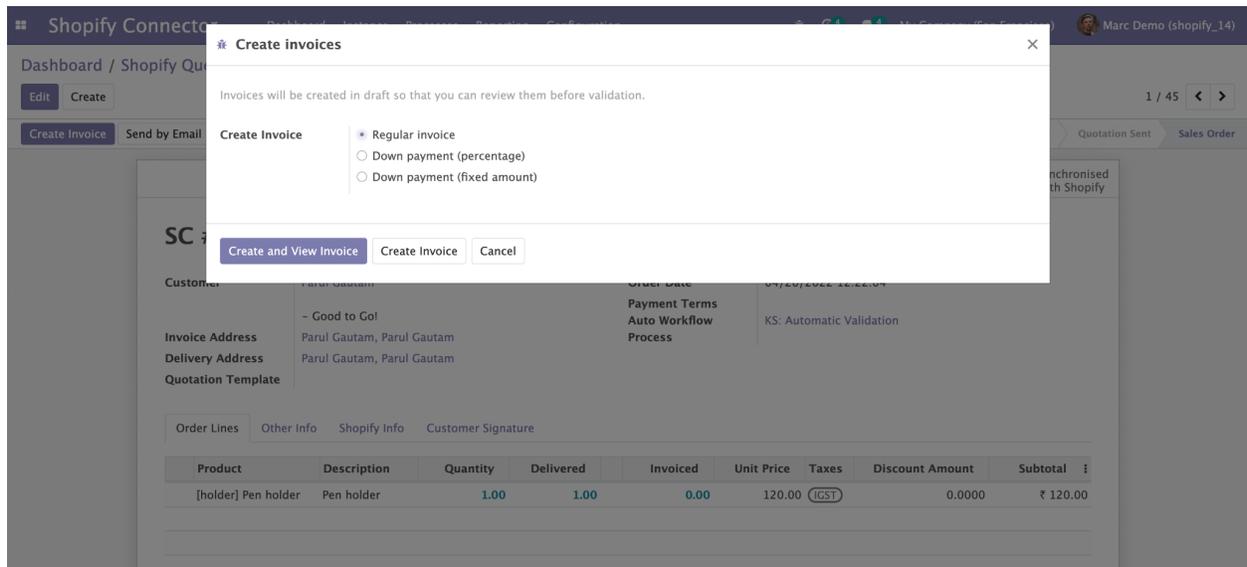
Filters Group By Favorites

Number	Order Date	Customer	Salesperson	Instance	Shopify Id	Next Activity	Company	Total	Shopify Status	Sync Stat...	Invoice Status
SC #401906162...	04/20/2022	Parul Gautam	Mitchell Admin	K_dev	4019061620793		My Company (San...	₹ 141.60	Completed		To Invoice
SC #401569405...	04/19/2022	Parul Gautam	Mitchell Admin	K_dev	4015694053433		My Company (San...	₹ 141.60	Paid		Fully Invoiced
SC #397383231...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3973832310841		My Company (San...	₹ 0.00	Paid		Fully Invoiced
SC #397483789...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974837895225		My Company (San...	₹ 23.60	Paid		Fully Invoiced
SC #397484376...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974843760697		My Company (San...	₹ 23.60	Pending		Fully Invoiced
SC #397484500...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974845005881		My Company (San...	₹ 23.60	Completed		Fully Invoiced
SC #397484605...	04/19/2022	Guest Customer	Mitchell Admin	K_dev	3974846054457		My Company (San...	₹ 23.60	Pending		Fully Invoiced
SC #397850954...	04/19/2022	Pihu	Mitchell Admin	K_dev	3978509549625		My Company (San...	₹ 21.00	Completed		Fully Invoiced
SC #401566066...	04/19/2022	Parul Gautam	Mitchell Admin	K_dev	4015660662841		My Company (San...	₹ 141.60	Paid		Fully Invoiced
S00044	04/18/2022	Parul Gautam	Mitchell Admin	K_dev	4014983643193		My Company (San...	₹ 1.15	Completed		Fully Invoiced

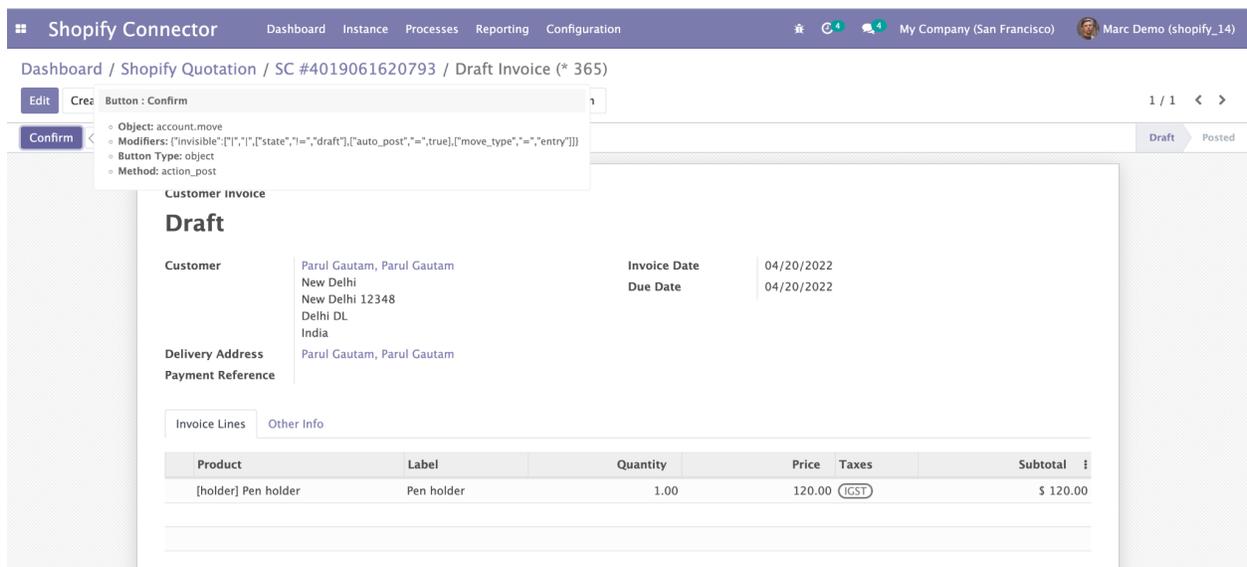
### e. Generate Invoice

Now to generate the invoice, click on the 'Click Invoice' button.

A dialog box will appear asking you to choose any of the three given options. Once you select the desired option, click on the 'Create and View Invoice' button.



Now click on the 'Confirm' button and complete the invoice.



## f. Creating Payment

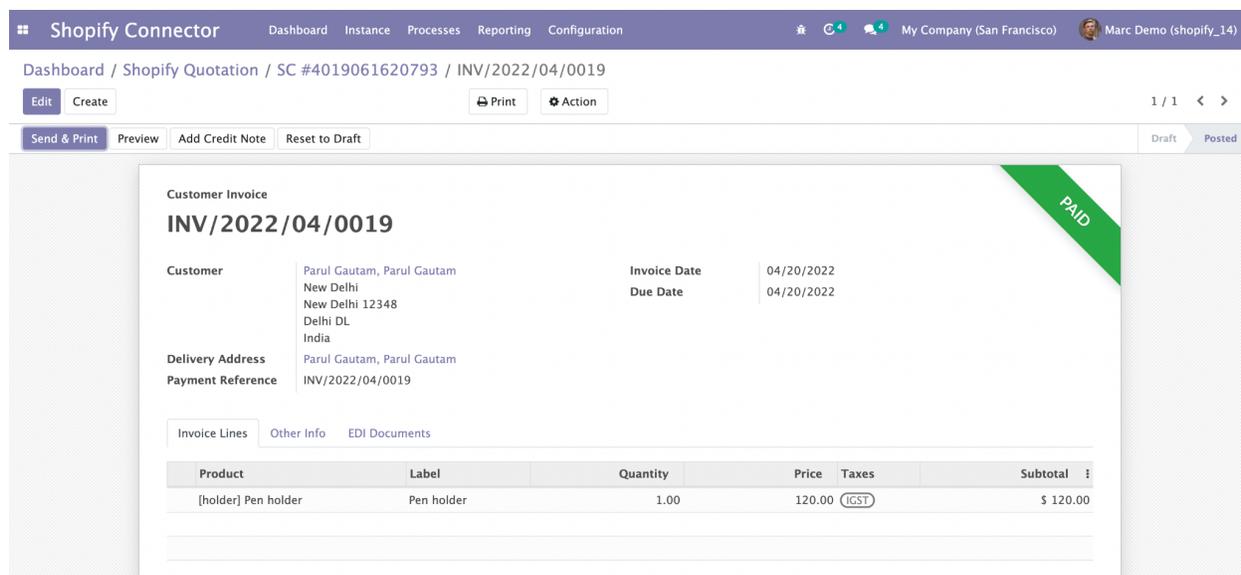
Go back to your order again and click on the 'Register Payment' button.

The screenshot shows the Shopify Connector interface. At the top, there's a navigation bar with 'Shopify Connector' and various menu items. Below that, the breadcrumb trail reads 'Dashboard / Shopify Quotation / SC #4019061620793 / INV/2022/04/0019'. There are buttons for 'Edit', 'Create', 'Print', and 'Action'. A secondary bar contains 'Send & Print', 'Register Payment', 'Preview', 'Add Credit Note', and 'Reset to Draft'. The 'Register Payment' button is highlighted. Below this, a modal window titled 'Button : Register Payment' displays technical details like 'Object: account.move', 'Context', 'Modifiers', 'Button Type: object', and 'Method: action\_register\_payment'. The main content area shows a delivery address for 'New Delhi' and a 'Payment Reference' of 'INV/2022/04/0019'. Below this is a table with columns for 'Product', 'Label', 'Quantity', 'Price', 'Taxes', and 'Subtotal'. The table contains one row: '[holder] Pen holder', 'Pen holder', '1.00', '120.00', 'IGST', and '\$ 120.00'.

Select the bank details and click on the 'Create Payment' button.

This screenshot shows the same interface as the previous one, but with the 'Register Payment' modal window open. The modal has a title bar with a close button. It contains the following fields: 'Journal' with a dropdown menu set to 'Bank'; 'Recipient Bank Account' with an empty dropdown; 'Amount' with a text input field containing '\$141.60'; 'Payment Date' with a dropdown menu set to '04/20/2022'; and 'Memo' with a text input field containing 'INV/2022/04/0019'. At the bottom of the modal are two buttons: 'Create Payment' and 'Cancel'. The background interface is dimmed, showing the same navigation and table as before.

Once the invoice is created you will see that your order status has been changed to PAID mode. (This is the complete procedure order creation–confirmation–delivery confirmation–invoice creation–confirmation and registering payment).

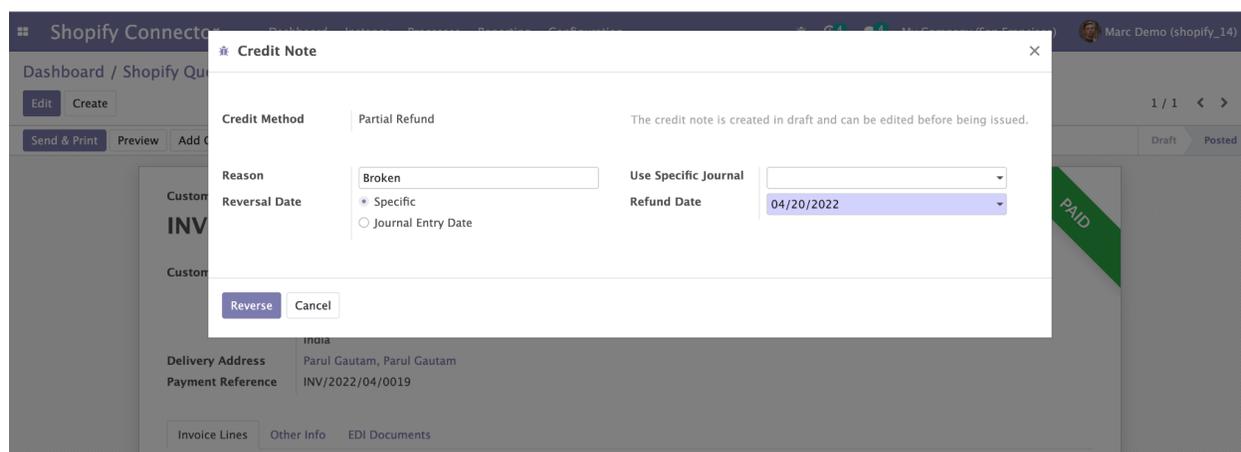


## g. Refund of Order and Product Stock

Many times it happens that the customer returns the order and needs a refund.

To return the order: Select the Add Credit Note button. You will get a window as given below where you have to add the reason for returning the order.

Once done click on the 'Reverse' button.



Click on the 'Confirm' button as shown below:

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793 / INV/2022/04/0019 / Draft Credit Note (\* 367) (Reversal of: INV/2022/04/0019, Broken)

Edit Create Print Action 1 / 1 < >

Confirm Preview Cancel Entry Draft Posted

**Customer Credit Note**  
**Draft**

**Customer** Parul Gautam, Parul Gautam  
New Delhi  
New Delhi 12348  
Delhi DL  
India

**Invoice Date** 04/20/2022  
**Due Date** 04/20/2022

**Delivery Address** Parul Gautam, Parul Gautam

**Payment Reference**

Invoice Lines Other Info

Product	Label	Quantity	Price	Taxes	Subtotal
[holder] Pen holder	Pen holder	1.00	120.00	IGST	\$ 120.00

You will find that two invoices will be created for the same order, one generated while confirming order & delivery and the second while returning the order. Click on the invoices and a window will appear like this:

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793 / Invoices

Create Upload

Search... Filters Group By Favorites 1-2 / 2 < >

**Company Data**  
Set your company's data for documents header/footer.  
Let's start!

**Invoice Layout**  
Customize the look of your invoices.  
★ Looks great!

**Create Invoice**  
Create your first invoice.  
Create

Number	Customer	Invoice Date	Due Date	Next Activity	Company	Tax Excluded	Total	Status	Payment Status
RINV/2022/04/0002	Parul Gautam, Parul Gautam	04/20/2022	Today		My Company (San Francisco)	\$ -120.00	\$ -141.60	Posted	Not Paid
INV/2022/04/0019	Parul Gautam, Parul Gautam	04/20/2022			My Company (San Francisco)	\$ 120.00	\$ 141.60	Posted	Paid

Open the invoice and click on the 'Refund in Shopify' button. ( This button would instantly initiate the refund process in Shopify)

The screenshot shows the Shopify Connector interface. At the top, there's a navigation bar with 'Shopify Connector' and various menu items like 'Dashboard', 'Instance', 'Processes', 'Reporting', and 'Configuration'. The user is logged in as 'Marc Demo (shopify\_14)'. Below the navigation bar, the breadcrumb trail reads: 'Dashboard / Shopify Quotation / SC #4019061620793 / Invoices / RINV/2022/04/0002 (Reversal of: INV/2022/04/0019, Broken)'. There are buttons for 'Edit', 'Create', 'Print', and 'Action'. A modal window titled 'Button : Reset to Draft' is open, showing details like 'Object: account.move', 'Modifiers: [{"invisible": [{"show\_reset\_to\_draft\_button": "=, false}]}]', 'Button Type: object', and 'Method: button\_draft'. The modal also displays customer and delivery address information. Below the modal, there's an 'Invoice Lines' section with a table:

Product	Label	Quantity	Price	Taxes	Subtotal
[holder] Pen holder	Pen holder	1.00	120.00	IGST	\$ 120.00

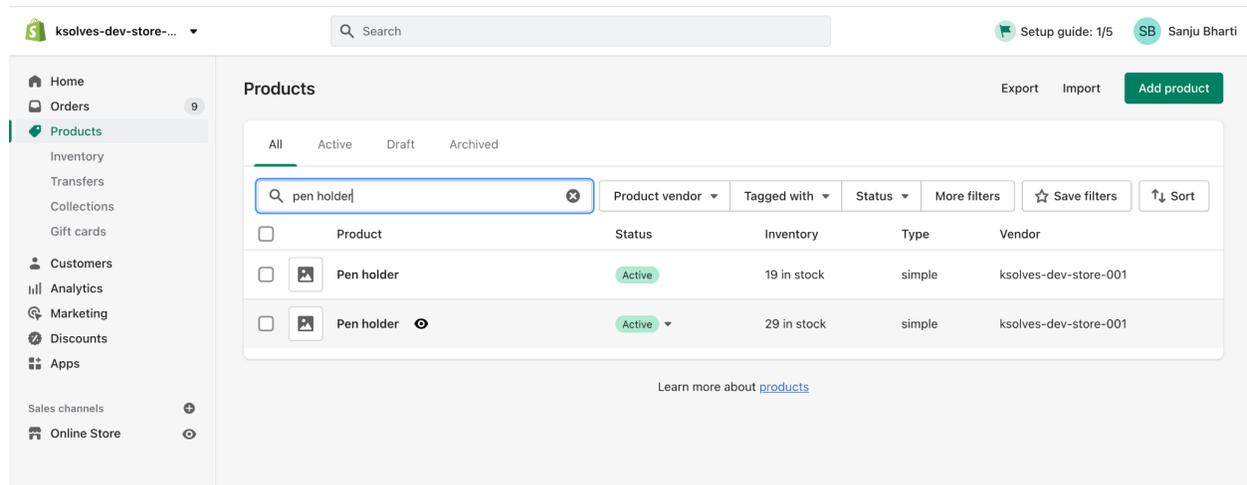
To match if your product stock has been updated at the Shopify end or not, go back to the Shopify backend.

Here you will see that your Order status has been changed from Paid to Refunded.

The screenshot shows the Shopify admin interface. The top navigation bar includes 'ksolves-dev-store...', a search bar, and 'Setup guide: 1/5' and 'Sanju Bharti'. The left sidebar has a menu with 'Home', 'Orders' (9), 'Drafts', 'Abandoned checkouts', 'Products', 'Customers', 'Analytics', 'Marketing', 'Discounts', and 'Apps'. The main content area shows an order #1027 with status 'Refunded' and 'Fulfilled'. The order was placed on '20 April 2022 at 11:52 am from Draft Orders'. The order details show 'Fulfilled (1) #1027-F1' with a product 'Pen holder' (SKU: holder) for ₹120.00 x 1. There's an 'Add tracking' button. The 'Notes' section shows 'No notes from customer'. The 'Customer' section shows 'Parul Gautam' with a link to '5 orders'.

Go to Products and search for your product.

Here the quantity of our product 'Pen holder' has been updated from 28 to 29 again.



This is how easily anyone can refund stock in Shopify from Odoo.

**Note:** To refund stock, it is important to import the product stock after importing the product from Shopify to Odoo(i.e to keep the stock of product updated at Odoo end as well).

## 11. Cancel Order

Now you can cancel a Sale Order from Odoo without even accessing your Shopify store.

**Note:** To cancel an Order in Odoo, the credit note must already be generated, and the order must have a reverse invoice.

Go to Dashboard>Orders>Select your order

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793

Edit Create Print Action 1 / 45

Create Invoice Send by Email Cancel Quotation Quotation Sent Sales Order

Button : Cancel

- Object: sale.order
- Modifiers: [{"invisible": true, "state": "not in", "draft": "sent", "sale": true}, {"id": "-", "false": true}]
- Button Type: object
- Method: action\_cancel

Customer: Parul Gautam  
 - Good to Go!

Invoice Address: Parul Gautam, Parul Gautam  
 Delivery Address: Parul Gautam, Parul Gautam  
 Quotation Template:

Order Date: 04/20/2022 12:22:04  
 Payment Terms: KS: Automatic Validation  
 Auto Workflow Process:

Order Lines Other Info Shopify Info Customer Signature

Product	Description	Quantity	Delivered	Invoiced	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	1.00	0.00	120.00	(IGST)	0.0000	₹ 120.00

Click on the Cancel button, you will get a dialogue box as given below, and click on the confirm button.

Shopify Connector Dashboard Instance Processes Reporting Configuration My Company (San Francisco) Marc Demo (shopify\_14)

Dashboard / Shopify Quotation / SC #4019061620793

Edit Create Print Action 1 / 45

Create Invoice Send by Email Cancel Quotation Quotation Sent Sales Order

Cancel Sales Order

Some products have already been delivered. Returns can be created from the Delivery Orders.

Button : Confirm

- Object: sale.order.cancel
- Button Type: object
- Method: action\_cancel

Customer: Parul Gautam  
 - Good to Go!

Invoice Address: Parul Gautam, Parul Gautam  
 Delivery Address: Parul Gautam, Parul Gautam  
 Quotation Template:

Order Date: 04/20/2022 12:22:04  
 Payment Terms: KS: Automatic Validation  
 Auto Workflow Process:

Order Lines Other Info Shopify Info Customer Signature

Product	Description	Quantity	Delivered	Invoiced	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	1.00	0.00	120.00	(IGST)	0.0000	₹ 120.00

Now click on the Cancel in Shopify button. (This button would instantly cancel the Sale Order in the Shopify store)

The screenshot shows the Shopify Connector interface for a quotation with ID SC #4019061620793. A modal dialog titled 'Button : Cancel in Shopify' is open, displaying the following configuration:

- Object: sale.order
- Modifiers: [{"invisible": true, "state": "not in", "cancel": true, "ks\_shopify\_status": "cancelled", "ks\_shopify\_order\_id": false}]
- Button Type: object
- Method: ks\_cancel\_sale\_order\_in\_shopify

The main interface shows the quotation details for customer Parul Gautam, with an expiration date of 04/20/2022 12:22:04. The order lines table is as follows:

Product	Description	Quantity	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	120.00	IGST	0.0000	₹ 120.00

Here you will see that your Sale Order has been cancelled successfully and the same will be updated over the Shopify store.

The screenshot shows the Shopify Connector interface after the quotation has been cancelled. The status bar at the top indicates the current state: Quotation > Quotation Sent > Sales Order > **Cancelled**. The summary bar shows:

- Customer Preview
- 1 Delivery
- 2 Invoices
- Synchronised With Shopify

The main interface shows the quotation details for customer Parul Gautam, with an expiration date of 04/20/2022 12:22:04. The order lines table is as follows:

Product	Description	Quantity	Unit Price	Taxes	Discount Amount	Subtotal
[holder] Pen holder	Pen holder	1.00	120.00	IGST	0.0000	₹ 120.00

**Important Note:** Synchronization between Odoo and Shopify becomes easy with the Ksolves connector. You can refer to our 120 days of FREE support on our Odoo Shopify Connector and take your eCommerce business to the next level.