Easy Products Auto Import - From Google Sheet

Importing products into Magento 2 from a Google Sheet can significantly streamline your inventory management and simplify bulk updates. This guide will walk you through the steps to set up and execute a product import from a Google Sheet to your Magento 2 store.

This powerful module allows you to manage products efficiently. Once installed, you only need to update your Google Sheet or the website to add or update products, regardless of when the changes are made. You can also manually set up a Cron job for automatic updates.

Features:

- Update Product Data in Google Sheets and Sync with Website.

- The admin can schedule a Cron job to add or update products through the system configuration settings.

- The admin can enable or disable the module via the system configuration settings.

- The admin can integrate Google Sheets using the system configuration settings.

- The admin can specify which products to add or update in Google Sheets through the system configuration settings.

Backend Configuration

1. **Import Enable**
   If you choose yes than auto import product module enable otherwise disable.

2. **Google Spread Sheet Id**
   Here you put your Google Spread Sheet Id. For id see given image.

3. **Google Sheet Id**
   Here you put your Google Sheet Id. For id see given image

4. Google Sheet Name

Here you put your Google Sheet Name. For name see given image in bottom, use name according google sheet id

5. Image Save Directory

Fill Directory name where you uploaded product image.

6. Upload JSON code

Cron configuration

Cron Setup   ⌄

Enable Cron
[store view]   Yes   ▾

Cron Run when Yes is selected

Set Cron
[store view]   In Every Hours   ▾

Hours
[store view]   14

Enter hours (numbers only 1 to 23)

Google Sheet File Structure

To understand the structure of the columns in the Google Sheet file for Magento 2, here are the detailed descriptions:

sku: A unique product identifier used as a key product attribute in Magento 2. It is required for all products and should be unique. The SKU can include both digits and letters without spaces, separated by underscores (_) and dashes (-).

store_view_code: The associated store view code. This column can be empty if the associated store or website has a single sub-item.

attribute_set_code: The product attribute set code. Ensure you create and configure the attribute set before importing products into Magento 2. It should match the attribute set name exactly, including capital letters (e.g., "Default", "Man shoes").

product_type: The type of the imported product. Use only lowercase letters (e.g., "simple", "configurable").

categories: Product categories in Magento 2 differ from those in Magento 1. Instead of comma-separated category IDs, Magento 2 requires the full names of assigned categories, including the full path, separated by a vertical bar (|). For example, "Default Category/Gear|Default Category/Gear/Bags" assigns the product to both Gear and Bags categories, with Bags as a subcategory of Gear.

product_websites: The associated product website code. Use lowercase letters only (e.g., "base").

name: The product name.

description: A product description.

short_description: A short description of the product. Both the description and short description can include valid HTML tags.

weight: The product weight. Use the format 1.00000.

product_online: Enables or disables the product. Use 1 to enable or 0 to disable.

tax_class_name: The product tax class. Use capital letters and spaces as in the Magento 2 backend (e.g., "Taxable Goods").

visibility: The product's visibility (e.g., "Catalog, Search", "Not Visible Individually").

price: The product price (e.g., 34.000).

special_price: The special price of the product (e.g., 34.000).

special_price_from_date: The start date for the special price (e.g., '2015-09-01 15:26:29').

special_price_to_date: The end date for the special price (e.g., '2015-09-01 15:26:29').

url_key: The URL key of the product. If this field is empty, a value is automatically generated based on the product name.

meta_title: The Meta title of the product.

meta_keywords: The product's Meta keywords.

meta_description: The product's Meta description.

base_image: The main product image and its path. Images should be uploaded to /pub/media/import. The path should follow this structure: /pub/media/import/sample_data/m/b/mb01-blue-0.jpg. You can also use a direct URL of an image, such as http://site.com/images/some_image.jpg.

base_image_label: The label of the base product image.

small_image: The name and path related to a small product image. Upload images to /pub/media/import. Follow this path structure: /pub/media/import/sample_data/m/b/mb01-blue-0.jpg. A direct URL of an image can also be used.

small_image_label: The label of the small product image.

thumbnail_image: The name and path related to a thumbnail product image. Upload images to /pub/media/import. Follow this path structure: /pub/media/import/sample_data/m/b/mb01-blue-0.jpg. A direct URL of an image can also be used.

thumbnail_image_label: The label of the product thumbnail image.

created_at: The date and time when the product was created. Use the format: yyyy-mm-dd hh-mm-ss (e.g., 2015-09-01 22:26:27). If the field is empty, the date and time of the database record are used.

updated_at: The date and time when the product was updated. Use the format: yyyy-mm-dd hh-mm-ss (e.g., 2015-09-01 22:26:27). If the field is empty, the date and time of the database record are used.

new_from_date: Sets the product as "new" from the specified date. Use the format: yyyy-mm-dd hh-mm-ss (e.g., 2015-09-01 22:26:27).

new_to_date: Stops displaying the product as "new" from the specified date. Use the format: yyyy-mm-dd hh-mm-ss (e.g., 2015-09-01 22:26:27).

display_product_options_in: Specifies the location on a product page where a block with options should be displayed (e.g., "Block after Info Column").

gift_message_available: Indicates that a gift message is available and will be displayed for the product.

custom_design: A custom design for the product page.

custom_design_from: The start date for the custom design of the product page.

custom_design_to: The end date for the custom design of the product page.

custom_layout_update: A custom XML layout for the product page.

page_layout: The product page layout (e.g., 1 Column). If this field is empty, the default product layout is used.

custom_options: A container for product options.

country_of_manufacture: The country of origin.

additional_attributes: Import of product custom options and data related to a simple product associated with a configurable product. Example for a simple product associated with a configurable product: "color=Red,has_options=0,required_options=0,size_pants=32" (Color attribute is "Red," the simple product has options: has_options = 0 – no required options, size pants attribute value is 32). Example for a configurable product: "has_options=1, required_options=1" (Product has required options, simple product SKU is associated in the _associated_sku column).

qty: The quantity of the product in stock.

out_of_stock_qty: The out-of-stock quantity of the product.

use_config_min_qty: Use the minimum quantity value from the config.

is_qty_decimal: Set to 1 if the quantity can be decimal.

allow_backorders: Set to 1 if backorders are enabled.

use_config_backorders: Use the default system value for backorder options (enable/disable).

min_cart_qty: The minimum required quantity of the product in the shopping cart.

use_config_min_sale_qty: Use the default config value for determining the minimum sale quantity of the product.

max_cart_qty: The maximum quantity of the product in the shopping cart.

use_config_max_sale_qty: Use the default config value to determine the maximum quantity of the product in the shopping cart.

is_in_stock: 1 if the product is in stock, 0 if the product is out of stock.

notify_on_stock_below: Set a minimum product quantity to start notifications about a low stock level.

use_config_notify_stock_qty: Use the default system value for the product's low stock notification.

manage_stock: Set to 1 to control the stock quantity of the product. Setting to 0 allows Magento 2 to consider the product as having an unlimited stock level.

use_config_manage_stock: Use the default system configuration value for stock management.

use_config_qty_increments: Use the product quantity increment from the current store configuration. Set to 1 to enable; set to 0 to disable.

qty_increments: The product quantity increment.

use_config_enable_qty_inc: Use the default store configuration value to enable the product quantity increment.

enable_qty_increments: Set to 1 or 0 to enable or disable the product quantity increment.

is_decimal_divided: Set to 1 if the product quantity increment can be decimal.

website_id: The associated product website ID.

related_skus: Related product SKUs separated by commas (e.g., 24-WG085_Group, 24-WG086, 24-WG083-blue, 24-UG01).

crosssell_skus: Cross-sell product SKUs separated by commas (e.g., 24-WG085_Group, 24-WG086, 24-WG083-blue, 24-UG01).

upsell_skus: Upsell product SKUs separated by commas (e.g., 24-WG085_Group, 24-WG086, 24-WG083-blue, 24-UG01).

additional_images: Additional product images (product media gallery) separated by commas. Images should be uploaded to /pub/media/import. Follow this path structure: /pub/media/import/sample_data/m/b/mb01-blue-0.jpg. A direct URL of an image can also be used.

additional_image_labels: Comma-separated labels for additional product images from the previous column.

update_product: Use 1 to add/update the product or 0 to not update.

update_image: Use 1 to add/update the image or 0 to not update.

Note: Use both columns (update_image, update_product) in the last sequence, and both are required.