

ConnectHooks

Complete User Guide

Step-by-Step Instructions for Every Feature

*ConnectHooks automatically receives data from your web forms
and saves it directly into your Salesforce records.
No copy-paste. No manual entry. Fully automatic.*

1. What is ConnectHooks?

ConnectHooks is a Salesforce application that bridges the gap between your web forms and Salesforce. When someone fills out a form on your website or a form platform (like Jotform, Typeform, or any other tool), ConnectHooks automatically captures that form data and saves it directly into the correct Salesforce record.

Think of it like this: without ConnectHooks, your team has to manually read each form submission and type the data into Salesforce. With ConnectHooks, this happens automatically within seconds of the form being submitted.

The Five Parts of ConnectHooks

ConnectHooks has five main sections. You will use them in this order:

Section	What It Is Used For
CH Config	One-time setup wizard. Run this first before anything else.
CH Webhook	Create and manage connections (webhooks) for each of your forms.
CH Mapping UI	Tell ConnectHooks which form fields go into which Salesforce fields.
Form Submission	See every form submission that ConnectHooks has received and processed.
Files	View files and documents that were attached to form submissions.

2. Installing the ConnectHooks Package

ConnectHooks is installed into your Salesforce organisation as a managed package. This chapter explains how to install it from start to finish.

2.1 Before You Install

Before you install ConnectHooks, make sure you have the following:

- A Salesforce organisation (Professional, Enterprise, Unlimited, or Developer edition).
- You are logged in as a System Administrator.
- You have a stable internet connection.
- You have the ConnectHooks installation link from the AppExchange or from the ConnectHooks team.

2.2 Installing from the Salesforce AppExchange or Installation Link

Step 1: Get the Installation Link

You will receive an installation link from the ConnectHooks team or find ConnectHooks on the Salesforce AppExchange. The link will look like a Salesforce URL starting with "<https://login.salesforce.com/packaging/installPackage.apexp...>" or a similar AppExchange URL.

What you will see on screen: A web browser opens with the Salesforce login page or the AppExchange page.

Step 2: Log into Salesforce

If prompted, enter your Salesforce username and password and click "Log In".

What you will see on screen: The Salesforce login screen. After logging in, you are taken to the package installation page.

Step 3: Choose Who to Install For

The installation screen shows three options:

- Install for Admins Only (Recommended for first install)
- Install for All Users
- Install for Specific Profiles

Select "Install for Admins Only" to start. You can give access to other users later.

What you will see on screen: A page titled "Install ConnectHooks" with three radio buttons. Select the first option.

Step 4: Click the "Install" Button

Click the blue "Install" button at the bottom of the page.

What you will see on screen: A loading spinner appears. The message "This may take a few minutes" is shown. Do not close the browser. Wait until installation is complete.

TIP: Installation usually takes 1 to 5 minutes. If it takes longer, Salesforce will email you when it finishes.

Step 5: Confirm Successful Installation

When installation is complete, you will see a green success message: "Package ConnectHooks was successfully installed."

What you will see on screen: A green success banner at the top of the page. Click "Done" to return to Salesforce.

TIP: If you see any errors during install, contact the ConnectHooks support team with a screenshot of the error.

2.3 Open the ConnectHooks App

Step 1: Click the App Launcher (the 9-dot grid icon)

In Salesforce, find the 9 small dots (grid icon) in the very top-left corner of the screen. Click on it.

What you will see on screen: The App Launcher panel slides out showing all available apps.

Step 2: Search for "ConnectHooks App"

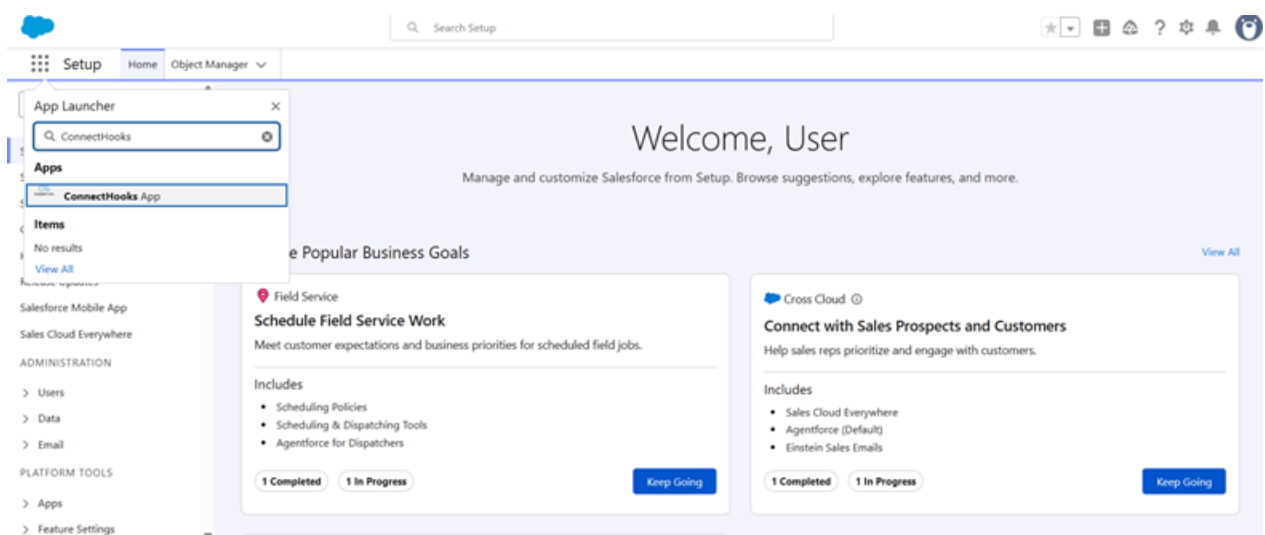
In the search box inside the App Launcher, type "ConnectHooks".

What you will see on screen: A result labelled "ConnectHooks" appears in the search results.

Step 3: Click on "ConnectHooks App"

Click the "ConnectHooks" result.

What you will see on screen: The ConnectHooks app opens. You will see a pink header bar with the app name "ConnectHooks" and five tabs across the top: CH Config, CH Webhook, CH Mapping UI, Form Submission, and Files.



3. First-Time Setup — CH Config Tab

Before ConnectHooks can receive any form submissions, you must complete a one-time setup. This is done through the "CH Config" tab and consists of 7 steps. Steps 1 to 4 must be done once and in order. Steps 5, 6, and 7 can be done at any time after Step 4.

IMPORTANT — Read This First Steps 1, 2, 3, 4 and 6 can ONLY be performed once and must be done in order. Steps 5, and 7 can be done in any order and can be changed at any time after Step 1,2,3,4 and 6 is complete.

3.1 Opening the Setup Wizard

Step 1: Click the "CH Config" Tab

In the ConnectHooks app, click on the "CH Config" tab in the top navigation bar.

What you will see on screen: A page loads showing a card titled "Instructions" and below it seven numbered steps. Each step is shown as a card with a title and either a button or checkbox.

3.2 Step 1 — Create a Public Site

A "Public Site" in Salesforce is a special web address that allows external tools (like your form platform) to send data to Salesforce without requiring a login. ConnectHooks needs this to receive form submissions.

Part A — Create the Public Site in Salesforce Setup

Step 1: Go to Setup (⚙)

Right-click the Salesforce Setup menu and open it in a new tab, or click the gear icon (⚙) → Setup.

What you will see on screen: The Salesforce Setup page.

Step 2: Search for "Sites" in the Setup Search Box

In the top-left search box of Setup, type "Sites". Under the results, click on "Sites" under the "User Interface" category.

What you will see on screen: The Sites list page appears. If you have existing sites, they are listed here.

Step 3: Click "New" to Create a New Site

Click the "New" button to start creating a new public site.

What you will see on screen: A form opens asking for site details.

Step 4: Fill in the Site Details

Complete the form with these required fields:

Site Label: Give it a clear name, for example "ConnectHooks Site"

Site Name: This auto-fills based on the label. Leave it as-is.

Default Web Address: Leave as auto-generated.

Active: Tick the checkbox to make the site active immediately.

Active Site Home Page: Type "UnderConstruction Page" (this is a system page, it is required).

Leave all other fields as their defaults.

What you will see on screen: The form with all your entries filled in.

Step 5: Click "Save"

Click the "Save" button.

TIP: Write down the full site URL. You will use it in Step 2 of the setup wizard.

The screenshot shows the 'ConnectHooks Site' configuration page in Salesforce. The page has a header with the 'SETUP Sites' logo. Below the header, the title 'ConnectHooks Site' is displayed. The main content area is titled 'Site Edit' and contains a form with the following fields:

- Site Label: ConnectHooks Site
- Site Name: ConnectHooks_Site
- Site Type: (empty)
- Site Description: (empty text area)
- Site Contact: User User
- Default Record Owner: User User
- Default Web Address: https://power-app-227-dev-ed.scratch.my.salesforce-sites.com/
- Active: ☒
- Active Site Home Page: UnderConstructionPage
- Inactive Site Home Page: (empty)
- Site Template: (empty)

At the top right of the form, there are 'Save' and 'Cancel' buttons.

Part B — Come Back to CH Config and Mark Step 1 as Done

Step 6: Go Back to the ConnectHooks App

Click back to the ConnectHooks app tab in your browser and make sure you are on the "CH Config" tab.

Step 7: Find the Step 1 Card

Look for the card labelled "Step 1: Create Public Site (Manual Step)".

What you will see on screen: A card with the title "Step 1" and the description "Create Public Site (Manual Step)". Inside the card is a checkbox labelled "Mark Done".

Step 8: Tick the "Mark Done" Checkbox

Click the checkbox labelled "Mark Done". This tells ConnectHooks you have completed the manual step.

What you will see on screen: The checkbox becomes ticked and the label changes to "Done". The Step 1 card now shows as complete. Step 2 below it becomes active (its button or dropdown is now enabled).

TIP: Only tick this checkbox after you have actually created the public site in Salesforce Setup. Ticking it without creating the site will cause errors in later steps.

Step 1

Create Public Site (Manual Step)

Done

Mark this step complete ONLY if you have already created a public site. You should follow the manual instructions outlined in the post-installation steps provided in the setup guide.

3.3 Step 2 — Set the Public Site to Use in ConnectHooks

Now you tell ConnectHooks which of your Salesforce public sites it should use to receive form data. You can only do this once — after you pick a site, it is locked in.

Step 1: Find the Step 2 Card

On the CH Config page, scroll to the card labelled "Step 2".

What you will see on screen: A card with title "Step 2" and a dropdown (combobox) labelled "Site to be used". If Step 1 is not yet complete, the dropdown is greyed out and the card shows a button labelled "Blocked" with the message "Please complete previous action!"

Step 2: Click the Dropdown — "Site to be used"

Click on the dropdown field labelled "Site to be used". A list of all your Salesforce public sites will appear.

What you will see on screen: A dropdown list showing the names of your available public sites.

Step 3: Select the Site You Created

Click on the site you created in Step 1 (e.g., "ConnectHooks Site").

What you will see on screen: The dropdown closes. The selected site name is shown in the field. The Step 2 card now shows a green completed state. The dropdown becomes read-only — you cannot change it again. Step 3 below becomes active.

TIP: Choose carefully — once you select a site here, it cannot be changed without re-installing the package.

The screenshot shows a configuration card titled "Step 2" with the subtitle "Set Public site to be used for exposing web service endpoint in ConnectHooks Application". Below the subtitle is a text instruction: "Select the site you want to use for ConnectHooks Integration in the Sites list, once you select it cannot be changed." On the right side of the card, there is a dropdown menu labeled "Site to be used" with a "Select Site" button and a downward arrow. Below the dropdown is a button labeled "ConnectHooks Site".

3.4 Step 3 — Assign Permissions to the Site Guest User

When your form platform sends data to Salesforce, it goes through the public site as a "guest" — a user who is not logged in. For security, this guest user needs specific permissions to read webhook settings and save form submission records. This step assigns those permissions automatically.

Step 1: Find the Step 3 Card

Scroll to the card labelled "Step 3: Assign 'ConnectHooks Public Site User Permissions' permission set to public site guest user".

What you will see on screen: A card with a button labelled "Assign Now" (in blue). If Step 2 is not complete, the button is replaced by a grey "Blocked" button with the message "Please complete previous action!"

Step 2: Click the "Assign Now" Button

Click the blue "Assign Now" button.

What you will see on screen: A spinning loading indicator appears on the button while Salesforce processes the request. Do not click anything else while you see the spinner.

TIP: This typically takes a few seconds. If it fails, check that you completed Steps 1 and 2 first.

Step 3: Wait for the Confirmation

Wait for the operation to complete.

What you will see on screen: The button changes to "Completed" and becomes grey (disabled). Step 3 is now marked as done. Step 4 below becomes active.



3.5 Step 4 — Create a Files Library

When forms are submitted with file attachments (like uploaded documents, images, or signatures), ConnectHooks needs a place in Salesforce to store them. This step creates a dedicated Files Library called "ConnectHooks Form Submissions".

Step 1: Find the Step 4 Card

Scroll to the card labelled "Step 4: Create Files Library to store all form response JSONs under that".

What you will see on screen: A card with a button labelled "Execute Now" (in blue). If Step 3 is not done, the button is grey and labelled "Blocked".

Step 2: Click the "Execute Now" Button

Click the blue "Execute Now" button.

What you will see on screen: A spinning loader appears on the button while Salesforce creates the library.

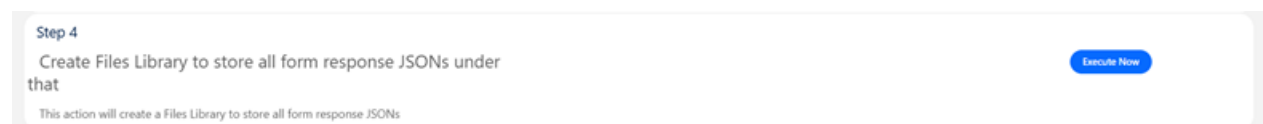
Step 3: Confirm the Library Was Created

Wait a few seconds for the process to finish.

What you will see on screen: The button changes to "Completed" and becomes grey. Step 4 is now done. Steps 5, 6, and 7 all become active at the same time (they can now be used in any order).

TIP: After this step completes, you can use Steps 5, 6, and 7 in any order you like.

SETUP MILESTONE Congratulations! Once Step 4 is complete, your ConnectHooks system is ready to receive form data. Steps 5, 6, and 7 configure optional and additional features.



3.6 Step 5 — Enable JSON Storage (Optional — Default is ON)

"JSON Storage" means ConnectHooks saves a complete copy of the raw form data alongside each submission record. This is very useful for checking what data was received and for debugging problems.

Step 1: Find the Step 5 Card

Scroll to the card labelled "Step 5: Enable JSON files storage under File Submission Record. (Default Enabled)".

What you will see on screen: A card with a checkbox labelled "Enable". The checkbox is ticked by default (enabled).

Step 2: Choose Your Setting

The checkbox is already ticked (enabled) by default. This means raw JSON data WILL be saved.

- Leave it ticked — to save the raw form data with each submission record.
- Un-tick it — if you do not want raw form data stored (saves storage space).

What you will see on screen: The checkbox toggles immediately when you click it. The setting saves automatically.

TIP: We recommend keeping this ON during your initial testing phase — it makes troubleshooting much easier.

TIP: If your forms contain sensitive personal data, review your data retention policy before enabling this.

Step 5

Enable JSON files storage under File Submission Record. (Default Enabled)

 Enable

3.7 Step 6 — Generate JSONPath API Key

The "JSONPath API Key" powers the smart field-path helper in the Mapping UI. When you are setting up field mappings later, this key allows the app to intelligently suggest the correct path to a field inside complex form data structures.

Step 1: Find the Step 6 Card

Scroll to the card labelled "Step 6".

What you will see on screen: A card with the description "Generate API Key using JSON for authentication purposes." and a blue button labelled "Generate API Key".

Step 2: Click "Generate API Key"

Click the blue "Generate API Key" button.

What you will see on screen: A spinner appears on the button while the key is being generated.

Step 3: Confirm the Key Was Generated

Wait for the process to complete.

What you will see on screen: A green success notification appears at the top of the screen: "API Key Generated Successfully!". The button becomes disabled. The key is stored securely and used automatically by the app.

TIP: You only need to do this once. The key is saved in the background and you do not need to copy or remember it.

Step 6

Generate API Key using JSON for authentication purposes.

Click the button below to generate the Authentication API Key.



3.8 Step 7 — Create Your First Webhook (Secure URL Code)

Step 7 is where you create your first webhook. A webhook is a unique secure connection that links one specific form to your Salesforce organisation. Each form you connect needs its own webhook.

Step 1: Find the Step 7 Card

Scroll to the card labelled "Step 7" (or the last step in the list). It shows a button labelled "New Webhook".

What you will see on screen: A card titled "Step 7: Create Secure URL Code" with a blue "New Webhook" button and a list of any existing webhooks below it. If Step 4 is not complete, the button is greyed out.



Step 2: Click "New Webhook"

Click the blue "New Webhook" button.

What you will see on screen: A pop-up window (modal) appears titled "New Webhook" with a form inside.

A screenshot of a modal form titled "New Webhook". It contains a text input field for "Webhook Name", two checked checkboxes for "Active" and "Enable JSON Storage", and "Cancel" and "Save" buttons at the bottom right.

Step 3: Fill in the Webhook Form

Complete the three fields in the pop-up:

Webhook Name (required):

Type a name for this webhook. Choose a name that describes which form it is for.

Example: "Contact_Form" or "Lead-Enquiry"

IMPORTANT: Do NOT use spaces. Use underscores (_), hyphens (-), or ampersands (&) instead.

Active (checkbox):

Leave this ticked (checked). This means the webhook is switched on and ready to receive submissions.

Enable JSON Storage (checkbox):

Leave this ticked. This saves a copy of the raw form data with each submission.

What you will see on screen: The modal form showing the three fields filled in.

Step 4: Click "Save"

Click the blue "Save" button at the bottom of the pop-up.

What you will see on screen: The modal closes. Your new webhook appears in the Step 7 card list. A green success message appears: 'Webhook "[YourWebhookName]" created successfully!' The webhook card shows: the Webhook Name, a Secure URL Code, and an API Key.

Step 7

New Webhook

Webhook Name
Contact_Form

☒ Active

☒ Enable JSON Storage

Secure URL Code
781rkjuc-e7dc7323-45gvi3-t1rn2e10-bwj26t1t-5po5gqme

Regenerate

API Key
78FuxY00RuL4pK2e1eobbbKf24V9gtuRL3DgvEIXeOo2W02B

Copy API Key

Regenerate API Key

Show Endpoint

https://power-app-227-dev-ed.acatch.my-ea1eaforce-estate.com/services/apexrest/webhook/ConnectHooksSubscriptionService/Contact_Form/781rkjuc-e7dc7323-45gvi3-t1rn2e10-bwj26t1t-5po5gqme

Copy

TIP: If you see an error "Spaces are not allowed in the webhook name", remove any spaces from the name and try again.

TIP: If you see a "Webhook Creation Limit Reached" message, you have reached the maximum number of webhooks for your plan. Contact ConnectHooks support to upgrade.

4. Managing Webhooks — CH Webhook Tab

The "CH Webhook" tab is where you manage all of your webhooks. A webhook is the secure connection point for each form. You can create new ones, activate or deactivate them, copy their credentials, and delete them from here.

4.1 Creating a New Webhook

You can also create new webhooks directly from the CH Webhook tab (not just from CH Config Step 7).

Step 1: Click the "CH Webhook" Tab

Click on the "CH Webhook" tab in the top navigation bar.

What you will see on screen: A list view showing all your existing webhooks. Each row shows the webhook name, active status, and creation date. If you have no webhooks yet, the list is empty.

Step 2: Click the "New" Button

Click the blue "New" button in the top-right area of the list.

What you will see on screen: A pop-up window titled "New Webhook" appears.

Step 3: Fill in the Webhook Details

Complete the form fields:

Webhook Name: Required. No spaces allowed. Use underscores or hyphens.

Example good names: "Website_Contact_Form", "Event-Registration", "Newsletter_Signup"

Example bad names: "Website Contact Form" (has spaces — not allowed)

Active: Leave ticked unless you want to create the webhook but not activate it yet.

Enable JSON Storage: Leave ticked to save raw form data for each submission.

What you will see on screen: The form with your entries.

Step 4: Click "Save"

Click the blue "Save" button.

What you will see on screen: The modal closes and the new webhook appears in the list. A green banner confirms: 'Webhook "[Name]" created successfully!'

TIP: After creating the webhook, click on it to open its detail page and copy the credentials.

4.2 Understanding the Webhook Record Screen

When you click on a webhook name, the webhook detail page opens. Here is a guide to everything you see on this screen:

Item on Screen	What It Means
Webhook Name	The name you gave this webhook when you created it. This cannot be changed after creation.
Active (checkbox)	When ticked — the webhook is ON and will accept form submissions. When un-ticked — the webhook is OFF and will reject all incoming form data.
Enable JSON Storage (checkbox)	When ticked — a copy of the raw form data is saved with each submission record. Un-tick to stop saving raw data.
Secure URL Code	A unique code used to authenticate your webhook URL. Your form platform needs this code to send data to this specific webhook. Shown in a grey box.

Regenerate (button next to Secure URL Code)	Generates a brand new Secure URL Code. The old code stops working immediately.
API Key	A 48-character secret password used for additional security when submitting form data. Your form platform needs this key too. Shown in a grey box.
Copy API Key (button)	Copies the API Key to your clipboard so you can paste it into your form platform.
Regenerate API Key (button)	Generates a brand new API Key. The old key stops working immediately.
Show Endpoint (button)	Reveals the complete webhook URL that your form platform needs to send data to. Click this to see and copy the full URL.
Delete icon (bin/trash icon)	Permanently deletes this webhook and all its mapping rules. Cannot be undone.

4.3 Copying Your Webhook Credentials

To connect your form to ConnectHooks, your form platform needs three pieces of information: the Endpoint URL, the Secure URL Code, and the API Key. Here is how to get all three.

Step 1: Open the Webhook Record

In the CH Webhook tab, click on the name of the webhook you want to connect to a form.

What you will see on screen: *The webhook detail page opens.*

Step 2: Copy the Secure URL Code

Find the "Secure URL Code" section. The code is shown in a grey box (it looks like a long string of letters and numbers). Click the copy icon next to it, or select all the text in the box and press Ctrl+C (Windows) or Cmd+C (Mac).

What you will see on screen: *A brief notification confirms the code was copied. Paste it into a document or directly into your form platform settings.*

Step 3: Copy the API Key

Find the "API Key" section. Click the "Copy API Key" button.

What you will see on screen: *A green success message appears: "API Key copied to clipboard!" Paste it where needed.*

Step 4: Get the Endpoint URL

Click the "Show Endpoint" button.

What you will see on screen: A box appears below the button showing the full endpoint URL. It looks like:

"https://yourcompany.my.site.com/services/apexrest/xccch/ConnectHooksSubmissionService/YourWebhookName/YourSecureCode"

Click the "Copy" button inside the box to copy the full URL.

TIP: Paste this Endpoint URL into your form platform as the webhook destination URL.

TIP: The Endpoint URL already includes your Secure URL Code inside it. Keep both for reference.

4.4 Viewing the Full Endpoint URL

The Endpoint URL is the complete web address that your form platform sends data to. It is built from four parts:

- Your Salesforce public site base URL (e.g., "<https://yourcompany.my.site.com>")
- The ConnectHooks service path: `/services/apexrest/xccch/ConnectHooksSubmissionService/`
- Your webhook name (e.g., `Contact_Form`)
- Your Secure URL Code (e.g., `ABC123XYZ...`)

The full URL example looks like:

https://yourcompany.my.site.com/services/apexrest/xccch/ConnectHooksSubmissionService/Contact_Form/ABC123XYZ

TIP Copy the full Endpoint URL using the "Show Endpoint" → "Copy" button. This is the URL you paste into your form platform as the webhook destination.

4.5 Activating and Deactivating a Webhook

Step 1: Open the Webhook Record

Click on the webhook name in the CH Webhook tab list.

Step 2: Find the "Active" Checkbox

Look at the left column. You will see a checkbox labelled "Active".

What you will see on screen: A checkbox that is either ticked (webhook is ON) or un-ticked (webhook is OFF).

Step 3: Click the Checkbox to Change the Status

If the webhook is ON and you want to turn it OFF: click the checkbox to un-tick it. A confirmation dialog will appear asking: "Do you really want to deactivate this?"

Click "OK" or "Yes" to confirm deactivation, or "Cancel" to keep it active.

What you will see on screen: *When deactivated, the checkbox is un-ticked and the webhook stops accepting form submissions immediately. When activated, the checkbox is ticked and the webhook begins accepting submissions immediately.*

TIP: Use deactivation (not deletion) when you want to temporarily stop receiving submissions — you can always reactivate it later without losing any settings.

4.6 Enabling or Disabling JSON Storage Per Webhook

Step 1: Open the Webhook Record

Click on the webhook name.

Step 2: Find the "Enable JSON Storage" Checkbox

In the left column, look for the checkbox labelled "Enable JSON Storage".

What you will see on screen: *A checkbox that is ticked (JSON is stored) or un-ticked (JSON is not stored).*

Step 3: Click to Toggle

Click the checkbox to turn JSON storage ON or OFF for this specific webhook.

What you will see on screen: *The checkbox updates immediately. The setting saves automatically — you do not need to click Save.*

TIP: Enabling JSON storage is recommended during initial setup and testing. Each stored JSON takes a small amount of Salesforce storage.

4.7 Regenerating the Secure URL Code

WARNING Regenerating the Secure URL Code will immediately make the old code invalid. Any form platform still using the old code will stop working until you update it with the new code.

Step 1: Open the Webhook Record

Click on the webhook name.

Step 2: Click the "Regenerate" Button

Next to the Secure URL Code field, click the "Regenerate" button.

What you will see on screen: *The old code is replaced with a new one in the grey box. A green success message appears: "Secure code regenerated successfully!"*

TIP: After regenerating, immediately update the new Endpoint URL in your form platform. The Endpoint URL changes because the Secure URL Code is part of it.

4.8 Regenerating the API Key

WARNING Regenerating the API Key makes the old key invalid immediately. Update your form platform with the new key right away.

Step 1: Open the Webhook Record

Click on the webhook name.

Step 2: Click "Regenerate API Key"

Find the "Regenerate API Key" button next to the API Key section. Click it.

What you will see on screen: *The old API Key is replaced with a new 48-character key. A green success message appears: "API Key regenerated successfully!"*

TIP: Copy the new API Key immediately using the "Copy API Key" button and update your form platform.

4.9 Deleting a Webhook

WARNING — CANNOT BE UNDONE Deleting a webhook permanently removes it and ALL of its mapping rules. Consider deactivating the webhook instead if you might need it again in the future.

Step 1: Open the Webhook Record

Click on the webhook name.

Step 2: Click the Delete Icon

Find the delete icon (a bin/trash icon or a button labelled "Delete") on the webhook card.

What you will see on screen: A confirmation dialog appears asking: "Do you really want to delete this webhook?"

Step 3: Confirm the Deletion

Click "OK" or "Yes" to confirm.

What you will see on screen: The webhook disappears from the list. A success message appears: 'Webhook "[Name]" has been Deleted.' All mapping rules for this webhook are also permanently deleted.

TIP: If you accidentally delete a webhook, you will need to recreate it and set up the mappings again from scratch.

5. Setting Up Field Mapping — CH Mapping UI Tab

"Field Mapping" tells ConnectHooks exactly what to do with the data it receives from a form. It answers the question: "When the form sends a field called 'first_name', which Salesforce field should that go into?"

You set this up once per webhook using the drag-and-drop Mapping UI. After that, ConnectHooks uses those rules automatically for every future submission.

5.1 Understanding the Mapping Screen Layout

The CH Mapping UI screen is divided into three areas:

Area	What It Shows
Top — Webhook Selector	A dropdown to choose which webhook you are configuring. You must select a webhook first before doing anything else.
Left Panel — Form Schema	Shows all the fields that come from your web form. After you upload a schema file, a tree of all form fields appears here. Each field has a name and a data type (text, number, etc.).
Right Panel — Salesforce Objects & Fields	Shows the Salesforce objects (like Lead, Contact, Account) where data will be saved, and all the available fields on each object.

TIP — Fullscreen Mode If the screen feels cramped, click the "Fullscreen" button (usually a four-arrows icon in the top-right corner). The mapping UI expands to fill your entire screen.

5.2 Selecting a Webhook

Step 1: Click the "CH Mapping UI" Tab

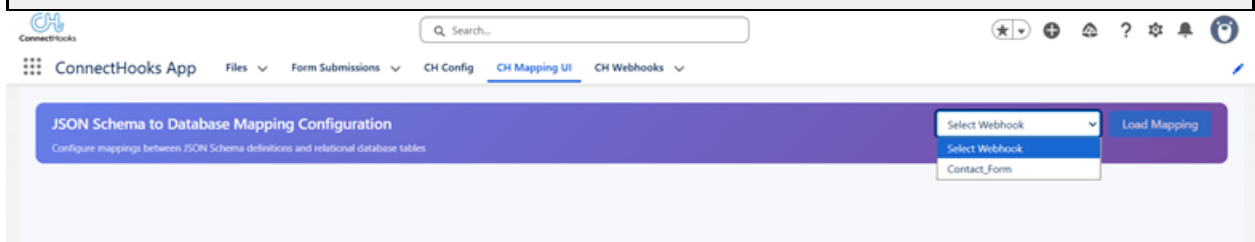
Click on the "CH Mapping UI" tab in the navigation bar.

What you will see on screen: The mapping interface loads. At the top you see a dropdown labelled "Select Webhook" or "Webhook". The left and right panels below it are empty initially.

Step 2: Choose Your Webhook from the Dropdown and Click Load Mapping

Click the webhook dropdown at the top of the screen. A list of your active webhooks appears. Click on the webhook you want to configure.

What you will see on screen: The selected webhook name appears in the dropdown. If this webhook already has mappings saved, they load automatically. A green success message may appear: "Webhook data loaded successfully."



TIP: Only active webhooks appear in this list. If you do not see your webhook, check that it is set to Active in the CH Webhook tab.

5.3 Uploading Your Json File

A schema file is a JSON file that describes the structure of your form — what fields it contains and what type of data each field holds. You need to upload this so ConnectHooks knows what fields your form sends.

How to get a schema file from your form platform:

- Download or export a "Sample Submission" or "Sample JSON" from your form tool.
- Some platforms call it "JSON Schema", "Webhook Payload", or "Sample Payload".
- The file must be a .json file. If your platform gives you a text payload, paste it into a text file and save it as .json.

Step 1: Find the Schema Upload Area

On the left panel of the Mapping UI, look for the upload area. It shows text like "Drop JSON Schema file here or" and a "Browse" button.

What you will see on screen: The upload area shows a dashed border box with a file icon and the upload instructions.

Step 2: Upload Your JSON File

You have two options:

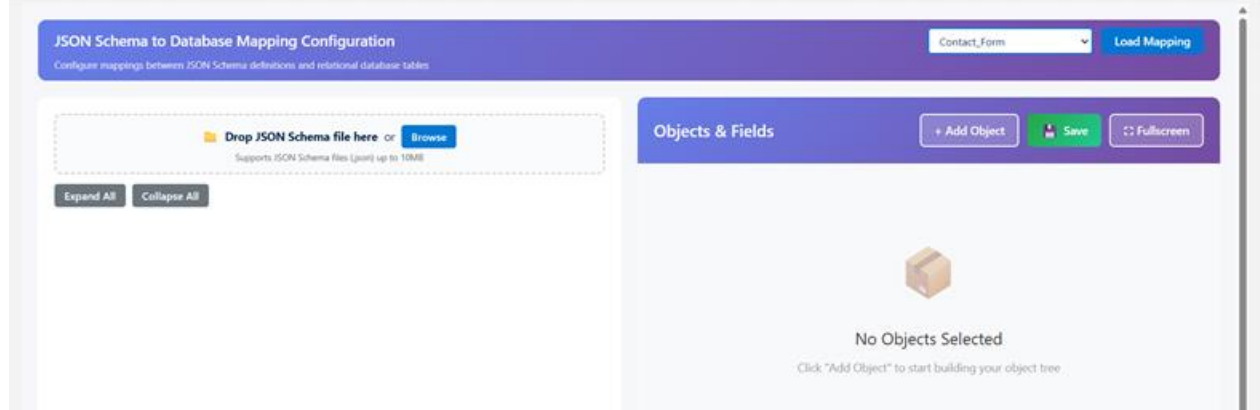
Option A — Drag and Drop:

Drag your .json file from your computer and drop it into the upload box.

Option B — Browse:

Click the "Browse" button. A file chooser window opens. Navigate to your .json file and click "Open".

What you will see on screen: After dropping or selecting the file, a loading process starts.



Step 3: Wait for the Schema to Load

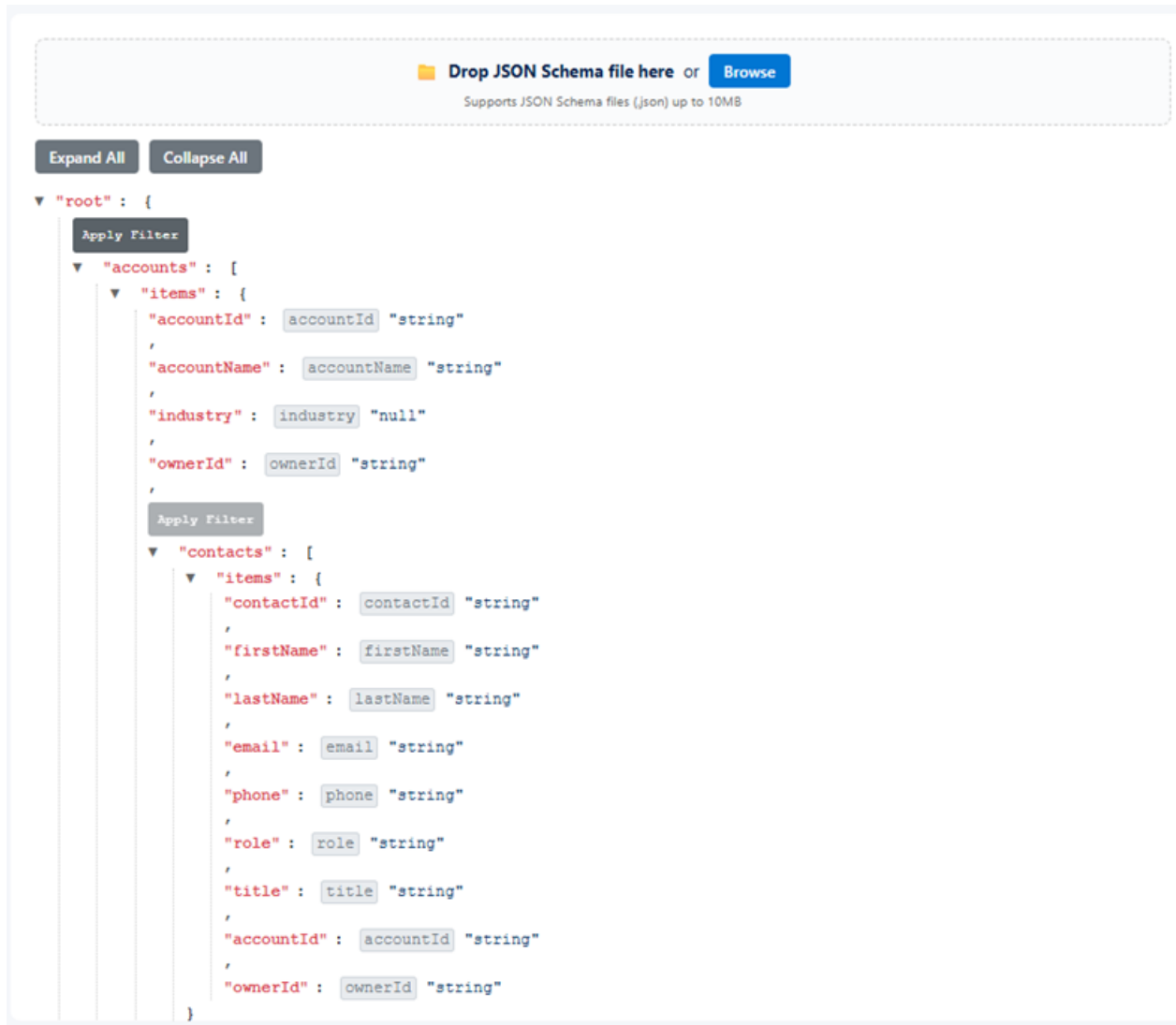
ConnectHooks reads and validates the JSON file automatically.

What you will see on screen: If the file is valid, the left panel fills with a tree structure showing all your form fields. A green success message appears: 'Schema file "[filename]" loaded successfully'

If ConnectHooks detects that your file is a raw JSON sample (not a proper schema format), it automatically converts it. An information message appears: "JSON data was automatically converted to JSON Schema."

TIP: The left panel tree shows all your form fields, each with its name and data type (string, number, boolean).

TIP: If you see "Schema could not be parsed", your file may be incorrectly formatted. Try downloading a fresh sample from your form platform.



5.4 Reading the Schema Tree — Left Panel

After uploading, the left panel shows all your form fields in a tree structure. Here is how to read it:

What You See	What It Means
A field name like "first_name"	A single text field from the form. You can drag this to a Salesforce field.
A field name with "string" label	This field contains text data (names, emails, addresses, etc.).
A field name with "number" label	This field contains a number (age, quantity, price, etc.).
A field name with "boolean" label	This field is a yes/no checkbox (true or false).

A field with a arrow icon	This is a group of related fields (an object). Click the arrow to expand and see the fields inside.
A field with a array icon	This is a repeating group (e.g., a list of products in an order). Special handling is needed for these — see the tips below.

5.5 Adding a Salesforce Object — Right Panel

A Salesforce "object" is like a database table — it is where records are stored. Common objects are Lead, Contact, Account, Opportunity, and Case or also add **custom objects**. You need to add the object(s) where your form data should be saved.

Step 1: Click the "+ Add Object" Button

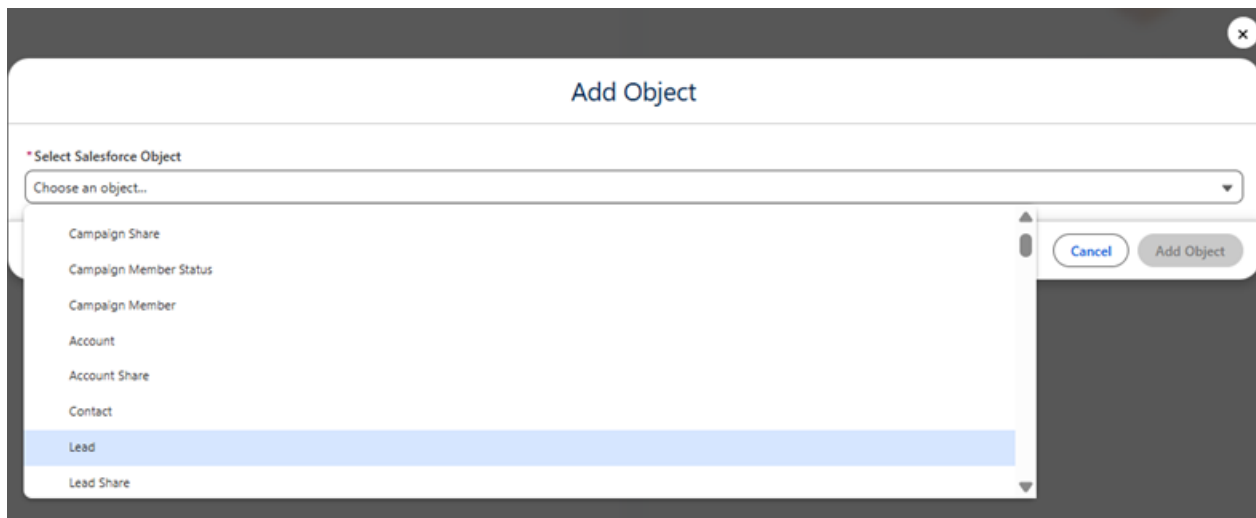
In the right panel, click the button labelled "+ Add Object" or "Add Object".

What you will see on screen: A popup window or slide-out panel opens showing a searchable list of all Salesforce objects available in your organisation (both standard and custom objects).

Step 2: Search for the Object You Need

Type the name of the Salesforce object you want to map to. For example, type "Lead" or "Contact".

What you will see on screen: The list filters to show matching objects as you type.



Step 3: Click the Object Name to Select It

Click on the object name in the list.

What you will see on screen: The popup closes. The selected object appears in the right panel as a card. The card shows the object name (e.g., "Lead") and a list of all its fields below it. A notification appears: "Required"

fields automatically added." — this means any fields that Salesforce requires to create a record are automatically highlighted.

TIP: You can add more than one object if your form data needs to go into multiple Salesforce objects. For example, form data might create both a Lead and a related Case at the same time.

TIP: If you try to add the same object twice, ConnectHooks highlights the existing one instead of adding a duplicate.

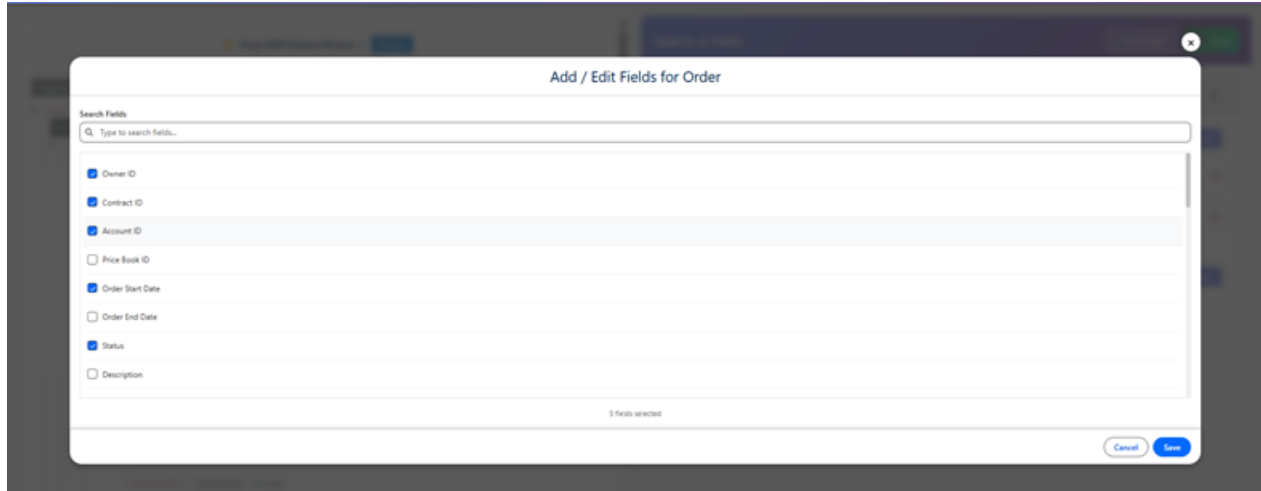
Step 4: Expand the Object to See Its Fields

Click on the object card or the expand arrow to see all the fields available on that object.

The screenshot shows a web interface titled "Objects & Fields" with a purple header. In the top right of the header are three buttons: "+ Add Object", "Save" (with a floppy disk icon), and "Fullscreen" (with a double arrow icon). Below the header, the "Lead" object is selected, indicated by a folder icon, the text "Lead", and a blue pill containing "2 fields". To the right of the object name is a trash can icon. Below this, a section titled "FIELDS" is expanded, showing two fields: "Last Name" and "Company". Each field has a dashed box placeholder containing the text "Droo JSON elements here", a left-pointing arrow, a right-pointing arrow, and a red "X" icon. To the right of the "FIELDS" section is a button labeled "Add / Edit Fields". Below the fields section is a section titled "CHILD RELATIONSHIPS" with a button labeled "+ Add Child Object". At the bottom of this section is a dashed box containing the text "No child objects added".

Step 5: Click the "Add/Edit Fields" Button

After click open a modal for field add multiple fields in one time.



What you will see on screen: A list of field names drops down below the object name. Each field shows its name and type (Text, Email, Phone, Date, Number, Checkbox, etc.).

Step 6: Click the "Add Child Object" Button

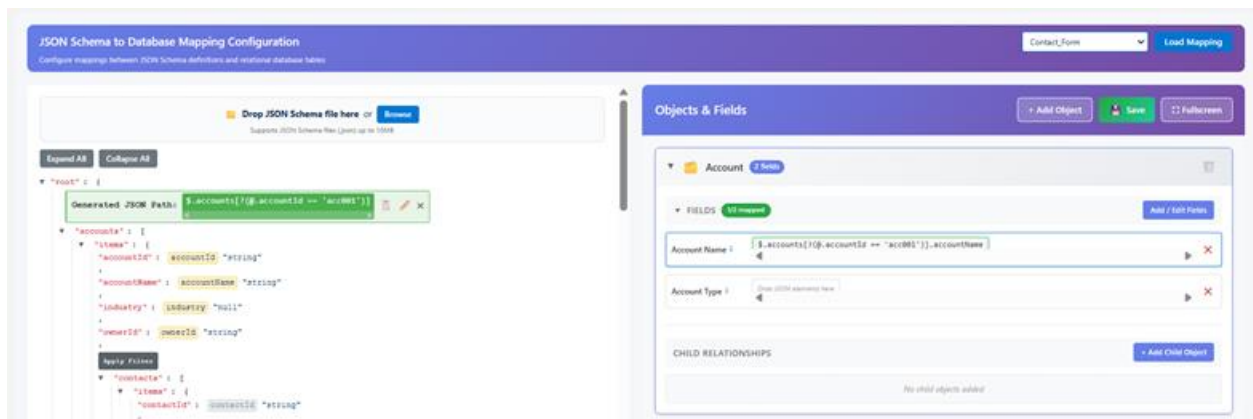
Our **Order__c** records belong to the **Account**. We model this parent-child relationship in the Object Tree.

Add Order__c as a child of Account

1. On the **Account** card, click + **Add Child Object**.
2. A modal appears. Select **Order__c** (or your custom order object).
3. Choose the **relationship field** — this is the lookup/master-detail field on Order__c that points to Account (e.g., **Account__c**).
4. Click **Add Child**.

5.6 Mapping Fields — Drag and Drop

Now the main work begins. You connect form fields (left panel) to Salesforce fields (right panel) by dragging a form field and dropping it onto a Salesforce field.



Step 1: Find the Form Field to Map

On the left panel, find the form field you want. For example, find "email" in the tree.

What you will see on screen: Each form field row has a drag handle on its left side (usually shown as two vertical dots or a grid of dots).

Step 2: Click and Hold the Form Field

Click and hold your mouse button on the form field row. Keep holding.

What you will see on screen: The row highlights and a drag preview appears following your mouse cursor.

Step 3: Drag Towards the Right Panel

While holding, move your mouse to the right panel where the Salesforce object and fields are shown.

What you will see on screen: As you hover over the right panel, compatible Salesforce fields highlight in green. Incompatible fields are shown in grey and cannot be dropped on.

Step 4: Drop onto the Target Salesforce Field

Move your mouse over the Salesforce field you want to map to (e.g., the "Email" field on the Lead object). Release the mouse button.

What you will see on screen: A new mapping appears. It shows:
Form field: email → Salesforce field: Email (Lead)
The mapping is added to the mapping zone and shown as a locked row.

TIP: Only compatible field types can be mapped. A text field from the form can map to Text, Email, Phone, or URL fields in Salesforce.

TIP: If a Salesforce field stays grey when you hover over it, the data types are not compatible. Choose a different Salesforce field.

TIP: You do not need to map every single form field — only map the ones you want saved in Salesforce.

5.7 Removing a Mapping

Step 1: Find the Mapping You Want to Remove

In the mapping zone, locate the mapping you want to delete.

Step 2: Click the Remove Button

Click the red "X" button or trash/delete icon on the mapping row.

What you will see on screen: *The mapping row disappears immediately.*

The screenshot shows a mapping interface with a header bar containing a dropdown menu labeled 'FIELDS', a green badge indicating '1/2 mapped', and a blue button labeled 'Add / Edit Fields'. Below the header, there are two mapping rows. The first row, labeled 'Account Name', contains a text input field with the JSON path `$.accounts[?(@.accountId == 'acc001')].accountName` and a red 'X' button to its right. The second row, labeled 'Account Type', contains a dashed box with the text 'Drop JSON elements here' and a red 'X' button to its right.

TIP: Removing a mapping does not affect any Salesforce records that were already created.

TIP: It only affects future form submissions — the removed field will no longer be saved.

TIP: Click Save after removing mappings to make the change permanent.

5.8 Working with Array Fields, Apply Filter & Update Filter

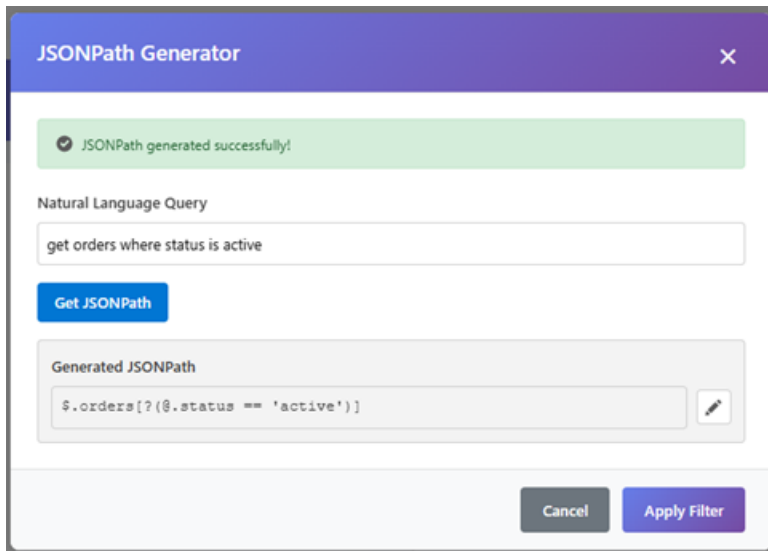
Arrays (like **orders** and **items**) can contain many rows. You need to tell ConnectHooks:

- **Which items to process** (optional filter)
- **Which Salesforce object each item maps to**

Step 1: Apply Filter (First Time)

In our example we only want to create Salesforce Order records for orders where `status = "active"`.

- In the Schema Panel, find the **orders** array node.
- Click the **Apply Filter** button]
- A modal dialog opens — **JSONPath Filter Editor**.



The screenshot shows a modal dialog titled "JSONPath Generator" with a close button (X) in the top right corner. Inside the dialog, there is a green success message: "JSONPath generated successfully!". Below this, there is a section labeled "Natural Language Query" with a text input field containing "get orders where status is active". A blue button labeled "Get JSONPath" is positioned below the input field. Underneath, there is a section labeled "Generated JSONPath" with a text input field containing "\$.orders[?(@.status == 'active')]" and a small edit icon (pencil) to its right. At the bottom of the dialog, there are two buttons: "Cancel" and "Apply Filter".

- The full generated path is shown below your input:

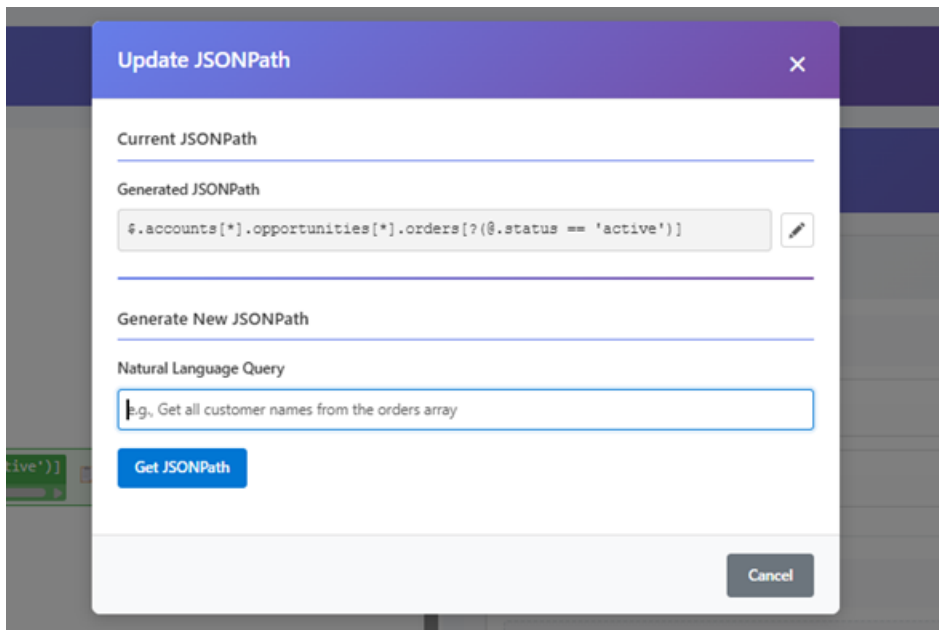
```
Generated JSON Path: $.orders[?(@.status=='active')]
```
- Click Apply Filter.
- The `orders` node in the Schema Panel now shows the filter badge:
`orders [?(@.status=='active')] ✎ ✕`

Step 2: Update Filter (Change an Existing Filter)

If you want to change the condition later (e.g., process `completed` orders instead):

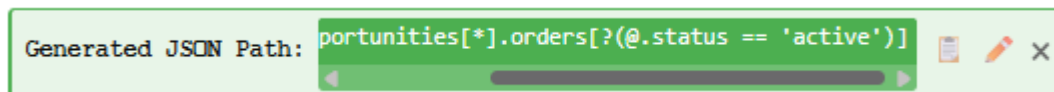
All JSON paths that use this filter (including nested child paths) are automatically updated throughout the mapping.

- Click the **pencil (✎) icon** next to the filter badge on the `orders` node.
- The filter editor modal opens with the current expression pre-filled.
- Change the expression to:
- `?(@.status=='completed')`
- Click **Apply Filter**.



Step 3: Remove a Filter

Click the **✕ (Remove)** icon next to the filter badge. The array reverts to `[*]` (process all items).



Step 4: Nested Array Filters (items inside orders)

You can also apply a filter on the nested `items` array:

- Expand the `orders` node in the Schema Panel.
- Find the `OrderLineItems [*]` node.
- Click **Apply Filter** on `OrderLineItems`.
- Enter: `?(@.unit_price > 50)`
- The generated path for items will now be:

```
$.orders[?(@.status=='active')].items[?(@.unit_price > 50)]
```

Notice how the parent filter (`?(@.status=='active')`) is automatically included — you do not need to type it again.

5.9 Using Fullscreen Mode

Step 1: Click the Fullscreen Button

Find the Fullscreen button in the top-right area of the CH Mapping UI. It looks like four outward-pointing arrows (expand icon).

What you will see on screen: *The mapping interface expands to fill your entire screen. The header and navigation are hidden to give you maximum working space.*

Step 2: Work in Fullscreen

Use the mapping UI normally — upload schema, add objects, drag and drop fields.

What you will see on screen: *All panels and tools are available in fullscreen mode.*

Step 3: Exit Fullscreen

Click the "Exit Fullscreen" button or the "← Return to Normal View" button at the top.

What you will see on screen: *The mapping UI returns to its normal size within the Salesforce page.*

5.10 Saving Your Mappings

IMPORTANT Your mappings are NOT saved automatically. You must click Save when you are finished. If you navigate away without saving, a warning message will appear asking if you want to leave.

Step 1: Review Your Mappings

Before saving, review all the mapping rows to make sure each form field is connected to the correct Salesforce field.

ConnectHooks runs a final validation:

- All required fields on every object must be mapped.
- Type compatibility is re-checked.
- Array filter consistency is verified (all fields on the same Salesforce object must use the same parent array filter).

What you will see on screen: *The mapping zone shows all connections. Double-check that required Salesforce fields are all mapped.*

Step 2: Click the "Save" Button

Click the "Save" button (usually in the top area of the right panel or screen).

What you will see on screen: A green success message appears: "Mappings saved successfully." The mappings are now stored in Salesforce and ConnectHooks will use them for all future submissions.

TIP: If the Save button is greyed out, you may have no mappings set up yet.

TIP: You can come back and edit mappings at any time — just select the webhook, make changes, and save again.

6.1 Test with Postman

Go to the **CH Config** tab in ConnectHooks, find your webhook **Eg.Contact-form**, and copy:

- API Key & Endpoint Url

API Key	sk_live_XyZ789
Endpoint URL	POST https://{your-org-domain}/services/apexrest/ConnectHooksSubmissionService/{webhookName}/{secureCode}

Step 1: Create a new request in Postman

- Open Postman ,Click **New** → **HTTP Request** → Set method to **POST**.
- Enter the EndPointURL:
POST <https://{your-org-domain}/services/apexrest/ConnectHooksSubmissionService/{webhookName}/{secureCode}>

Step 2: Click the Headers tab and add:

Key	Value
Content-Type	application/json
x-api-key	sk_live_XyZ789

IMPORTANT :- You can provide the API Key in either the Header or the Body. Ensure you add the API Key in only one place—not both.

Step 3: Add the Request Body

- Click the **Body** tab → select **raw** → select **JSON** from the dropdown.
- Paste the Youre Mapped JSON.
- Alternative API Key placement: You can also add the API Key at the inner root level of your JSON body like this:

```
• {  
    "x-api-key": "sk_live_XyZ789",  
    "your_mapped_elements": "...",  
    "data_array": []  
}
```

- Click [Send](#)

Step 4: Expected Response — Success

HTTP Status: **200 OK**

```
{  
  "success": true,  
  "webhookName": "shop-orders",  
  "recordsCreated": 4,  
  "details": {  
    "Account": 1,  
    "Order__c": 1,  
    "Order_Item__c": 2  
  },  
  "skipped": {  
    "reason": "Filter did not match",  
    "orders": ["ORD-002"]  
  },  
  "timestamp": "2024-01-15T10:30:00.000Z"  
}
```

Json path creator / New Request

POST https://power-app-227-dev-ed.scratch.my.salesforce-sites.com/services/apexrest/xccch/ConnectHooksSubmissionService/Contact_Form/781rkj...

Send

Docs Params Authorization Headers (10) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```

267     "title": "Security Manager",
268     "accountId": "acc003",
269     "ownerId": "user001"
270   },
271 ],
272   "opportunities": [
273     {
274       "opportunityId": "opp004",
275       "opportunityName": "Security Audit and Compliance",
276       "stage": "Closed Won",
277       "stageName": "Closed Won",
278       "amount": 95000,
279       "closeDate": "2024-03-20",
280       "ownerId": "user001",
281       "events": [

```

Body Cookies (3) Headers (12) Test Results 200 OK · 1.92 s · 739 B Save Response

{ } JSON Preview Visualize

```

1 [{"warnings":["No field values extracted for mapping: Order"],"timestamp":"2026-03-02T12:18:51.983Z","recordIds":
  [{"0019I00000PaLlyQAF","0019I00000PaLlzQAF","0019I00000PaLm0QAF","0019I00000PaLm1QAF","0019I00000PaLm2QAF",
    "0019I00000PaLm3QAF","0069I00000Gm54DQAR","0069I00000Gm54EQAR","0069I00000Gm54FQAR","0069I00000Gm54GQAR",
    "0069I00000Gm54HQAR","0069I00000Gm54IQAR","0069I00000Gm54JQAR"],"recordsCreated":13,"webhookName":"Contact_Form",
    "status":"success","success":true}]}

```

Step 5 : Check You're Salesforce Org

- **Open the App Launcher:** Look at the top-left corner of your Salesforce screen. Click on the icon that looks like 9 small dots (the App Launcher).
- **Search for Your Object:** In the search box that appears, type the name of the object you chose during mapping (for example: type "Leads" or "Contacts").
- **Go to the Record List:** Click on the name of your object in the search results.
- **See Your Data:** Once the page opens, make sure your view is set to "All Records" or "Recently Viewed." You will now see your brand new record sitting right there in the list!

	Opportunity Name	Account Name	Stage	Close Date	Am...	Opportun...	Created Date
1	Automated Inventory System	Manufacturing Solutions Inc.	Closed Won	15/06/2024		User	02/03/2026, 5:48 pm
2	Cloud Migration	Demo1	Negotiation	15/03/2024		User	02/03/2026, 5:48 pm
3	E-commerce Platform Implementation	Retail Excellence Group	Qualification	05/04/2024		User	02/03/2026, 5:48 pm
4	IT Infrastructure Upgrade	Innovative Solutions Ltd.	Closed Won	01/12/2023		User	02/03/2026, 5:48 pm
5	Patient Data System Upgrade	Healthcare Innovations Group	Proposal	01/05/2024		User	02/03/2026, 5:48 pm
6	Security Audit and Compliance	Financial Dynamics Corp	Closed Won	20/03/2024		User	02/03/2026, 5:48 pm
7	Website Redesign Project	Demo1	Proposal	10/02/2024		User	28/02/2026, 5:28 pm
8	Website Redesign Project	Demo1	Proposal	10/02/2024		User	28/02/2026, 5:27 pm
9	Website Redesign Project	Demo1	Proposal	10/02/2024		User	28/02/2026, 5:23 pm
10	Website Redesign Project	Demo1	Proposal	10/02/2024		User	28/02/2026, 5:29 pm
11	Website Redesign Project	Demo1	Proposal	10/02/2024		User	02/03/2026, 5:48 pm

6. Viewing Form Submissions — Form Submission Tab

Every time ConnectHooks receives a form submission, it creates a record in the "Form Submission" tab. This is your complete audit trail — you can see every submission, whether it succeeded or failed, and what data was received.

6.1 Opening the Submissions List

Step 1: Click the "Form Submission" Tab

Click on the "Form Submission" tab in the ConnectHooks navigation bar.

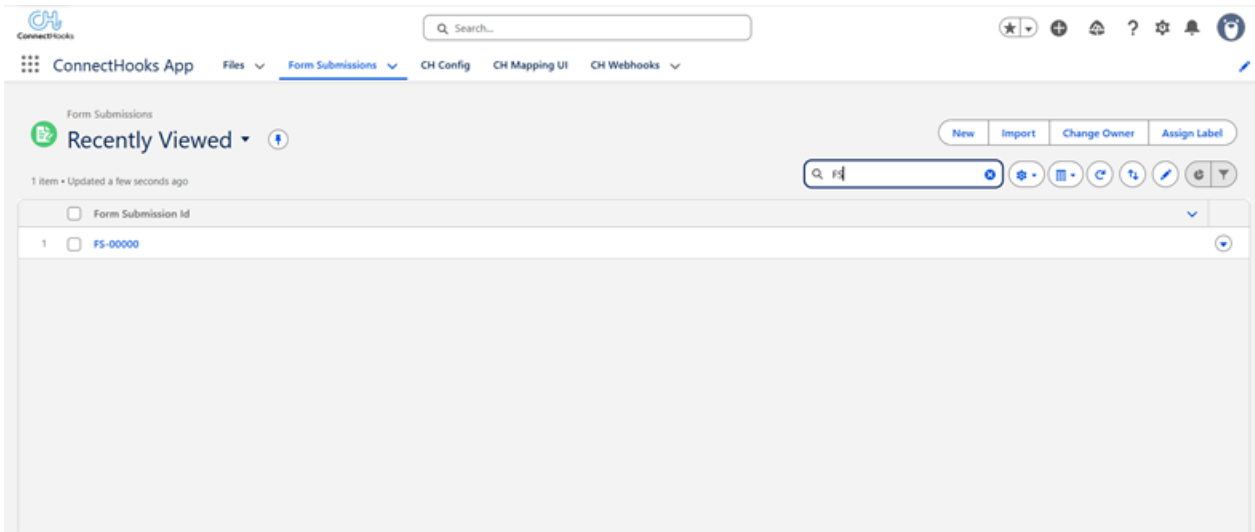
What you will see on screen: A list view appears. Each row represents one form submission. The list shows columns for: Submission Number (e.g., FS-00001), Form Name, Status, and Date/Time.

Step 2: Search the List

You can scroll through the list, or use the search bar at the top to search by form name, date, or submission ID.

What you will see on screen: The list updates as you type in the search box.

TIP: Submissions are ordered with the most recent first. If you just submitted a test form, it should appear at the top.



6.2 Reading a Submission Record

Step 1: Click on a Submission Row

Click on any submission in the list to open its full detail record.

What you will see on screen: The submission detail page opens. It shows all the information for that one submission.

Here is what each field on the submission record means:

Field Name	What It Shows
Submission Number	A unique auto-number ID for this submission (format: FS-00001, FS-00002, etc.). Use this when reporting a specific issue.
Form Name	The name of the form that was submitted. Comes from the form platform.
Form Entry ID	The unique ID assigned by your form platform to this specific submission.
Form Entry Link	A clickable link back to the original form submission on your form platform (if provided).
Status	"Success" = the data was saved to Salesforce successfully. "Failed" = something went wrong. See Error Details for the reason.
SObject Record ID	The ID of the Salesforce record that was created or updated by this submission. Click it to open that record.
Error Details	If the status is "Failed", this field shows the specific error message explaining what went wrong.
Raw JSON	The complete raw data received from the form (only shown if JSON Storage is enabled). Useful for verifying exactly what data was sent.

Submission Unique ID	A unique identifier used internally to prevent the same submission being processed twice.
-----------------------------	---

7. Files and Attachments — Files Tab

If your web forms allow users to upload files (documents, photos, PDFs, signatures, etc.), ConnectHooks automatically downloads those files and attaches them to the Salesforce record that was created by the form submission.

7.1 Viewing Files on a Salesforce Record

Step 1: Open the Salesforce Record

Navigate to the record that was created by the form submission. (You can get there by clicking the SObject Record ID from the Form Submission record.)

What you will see on screen: *The record detail page (e.g., a Lead page).*

Step 2: Scroll Down to the "Files" Related List

On the record page, scroll down until you see a section called "Files" or "Notes & Attachments". This section shows all files attached to this record.

What you will see on screen: *A list of file names. Each file shows its name, type (PDF, JPEG, etc.), size, and upload date.*

Step 3: Click a File to Open or Download It

Click on any file name to open it (images and PDFs open in the browser) or download it.

TIP: If you cannot see the Files related list on the record page, you may need to scroll further down, or ask your Salesforce administrator to add the Files related list to the page layout.

7.2 Viewing All Files in the Files Tab

Step 1: Click the "Files" Tab

In the ConnectHooks app, click the "Files" tab.

What you will see on screen: *A list of all files that ConnectHooks has received and stored, from all webhooks and all submissions.*

Step 2: Search or Upload for a File

Use the search bar to find a specific file by name, or browse through the list.

What you will see on screen: *The list shows file names, associated records, and dates.*

Step 3: Click a File to Open It

Click on any file name to open or download the file.

TIP: All files are stored in a Salesforce Files Library called "ConnectHooks Form Submissions". You can also access this library from the Salesforce "Files" app (the standard Salesforce Files tab).