

The purpose of this document is to show you that you can use DocMergy using the basic native HubSpot properties based on a **deal trigger**

You can download this document and put into the DocMergy to see how it works.

You will just need to pre fill the properties below which can just be made up

Document structure

- Basic Tokens
- Get Associated Objects
- Search by Unique IDs
- If Branches
- If Block Branches
- Dynamic Images
- Dynamic Hyperlinks

Required properties that need to be prefilled in order for this work

Deal Name

Deal Amount

Deal Beginning Date

Deal Close Date

Deal Description

A Contact

 Contact First Name

 Contact Email

Have 2 contacts associated with a deal

Line Item Name

Line Item Description

Need to create a custom property called: 'Product Image URL' with the internal name being 'product_image_url' and the token being {{ object.product_image_url }}

Have a deal in the Appointment Scheduled Stage

Have Deal Collaborator empty (no value in the property)

Basic Tokens

Basic native HubSpot property tokens

Deal name: Contract Renewal

Deal name in bold: **Contract Renewal**

Deal name in capitals: CONTRACT RENEWAL

Deal Amount: 1200

Deal Beginning Date: 18 August 2024

Deal Close Date: 12:21AM. 01 Dec 24

Deal Description: This is a test description

1. Basic Tokens with styling

Deal name: Contract Renewal

Deal name in bold: **Contract Renewal**

Deal name in bold upper case: **CONTRACT RENEWAL**

2. Basic Tokens with formatting (Date Formatting & Token Generator)

Deal Close Date normal: 12:21AM. 01 Dec 24

Deal Close Date (eg. 13 April 2025: 01 December 2024

Deal Close Date(eg. Sun, 13 April 25: Sun, 01 Dec 24

Close Date (eg. 13/04/2025): 01/12/2024

3. Basic Tokens in text boxes

Deal Amount

1200

Deal Beginning Date: 18 August 2024

Deal Name

Contract Renewal

Deal amount

1200

Deal Begin Date: 18 August 2024

Deal name

Contract Renewal

4. Basic Tokens in header & footer

This deal is: Contract Renewal

- Double click the heading or footing section and place the token there

Get Associated Objects

1. Get singular version

- The code gets the **first contact associated with the deal** and returns their **first name**

Someone

The function `get_associated_objects` takes 4 arguments

1. "contact": is the source object of the association
 2. `object.hs_object_id`: the id of the source object, here it is the enrolled object ID
 3. "deal" the associated object type you want to get data from
 4. The last argument is "order_by" and is the property you want to use for ordering the list of "deals". You can get more info here <https://support.integrationlue.com/docmergy-how-to-get-data-from-associated-objects>
- The `[0]` tells DocMergy to only grab the first associated deal you can here use `[1]` if you want the second one or `[-1]` if you want the last one
 - Dealname is the internal property name of the "deal" you want to display
 - The `{{ }}` around it are telling docmergy to render the property in the document

2. For loop version

- For every contact associated with the current deal, show their first name and email address

First Name	Email
John	newnic12@gmail.com
Someone	test123@gmail.com

3. For loop images

- This code loops through all line items associated with a deal and displays the line item name, description as well as the image and image text

Title	Description	Image
Line Item Test 1	This is the test 1	<div>This Image:  Line Item Test 1</div>
Line Item Test 2	This is line item test 2	<div>This Image:  Line Item Test 2</div>

Search By Unique IDs

1. Get singular version

- This shows the deal name of the with the HubSpot ID: 32226481156

Search for the unique ID of and show me the deal name: Contract Renewal

2. For loop version



- Loop through two specific deals, identified by their HubSpot object IDs (32226481156 and 35640046409), getting the deal name, deal amount and deal close date and store each one in the variable c for use inside the loop.

Deal Name	Deal Amount	Deal Close date
Contract Renewal	1200	12:21AM. 01 Dec 24
Test Deal	2500	10:19PM. 14 Apr 25

- `search_by_unique_ids("deal", [...])` pulls full data for the listed deal IDs.
- The for loop goes through each one and refers to it as c.
- Inside the loop, you can access things like dealname, amount, ect

3. For loop images

- This loop grabs two specific deals by their HubSpot IDs and displays each deals name, amount and converting the 'product picture URL' to a visible image
- You will need to create a custom property for this called: 'Product Image URL' with the internal name being 'product_image_url'
- Upload some images to HubSpot files, grab the URL and paste it into the property on the two deals. To see the demo get two random images and put each URL on two different deals.
- Put the token {{ c.product_image_url }} in the image alt text below. It needs to be "c." so its taking the other deals URL and not just the deal that is enrolled

Deal Name	Deal Amount	Profile Picture
Contract Renewal	1200	 Workflow Calculation Actions <small>By Integration Glue 100+ installs</small> Create advanced calculations, including date and time calculations
Test Deal	2500	 Workflow Associations Manager <small>By Integration Glue</small> Automatically create & delete HubSpot association & association labels <small>● Installed</small>

If Branches

1. Basic If Branch

- If dealname has a value, it shows the message.
- If it's blank or missing, it shows nothing at all

Hello! You're working on the Contract Renewal deal.

2. If Else Branch

- This code checks if the **deal has a name**. If it does, it shows a message with the deal name; if not, it shows a default message saying the deal doesn't have a name yet.

Hello! You're working on the Contract Renewal deal.

3. If Elif Branch

- This code checks the dealstage of a deal
- If it's "Appointment Scheduled," it shows one message;
- If it's "Closed Won," it shows a different message
- Won't show anything if neither of those are recognized

This deal is in the Appointment Scheduled stage

4. If Elif Else Branch.

- This code checks the dealstage of a deal
- If it's "Appointment Scheduled," it shows one message;
- If it's "Closed Won," it shows a different message
- If it's anything else it will show: This deal is in a different stage

This deal is in the Appointment Scheduled stage

If Block Branch

1. Examples for normal text

- If the dealname property **has a value**, it shows the message. If it's empty or missing, nothing is shown.

Hello! You're working on the Contract Renewal deal.

2. Examples for in tables

- If the dealname property exists and has a value, it shows the deal's name followed by its description.
- If dealname is blank, nothing is shown.

Deal Name and Description	Deal Start Date
Purchase Agreement – Purchase agreement for the product	10 January 2023
Contract Renewal - This is a test description	18 August 2024

3. Examples for deleting images, tables and text boxes

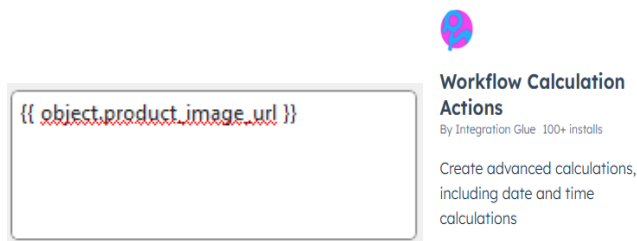
- The code is setup so If the token contains no value, delete everything between `{% block if object.hs_all_collaborator_owner_ids %}` and `{% end block %}`
 - If the token **does** contain a value, then keep everything in between.
 - In this case the property for 'Deal Collaborator' is **empty** so when you run it through DocMergy there will be nothing shown

Dynamic Images

- Dynamic images use a placeholder but will take an image URL on the custom property on the enrolled object

Eg. Display Product Picture

- Eg. With the custom property you made for the product image, put the token in the ALT text as seen below, this allows it to change dynamically



Dynamic Hyperlinks

- Dynamic hyperlinks use a placeholder of any text of your choice but will take the property with a URL (could be a website) associated with the enrolled object and change according to the enrolled object.
- If the contact has a property with a URL in it then the Dynamic Hyperlink grabs that link and puts it in the text due to the token in the hyperlink
- This URL will take you to your product image on the deal

- Address: `{{ object.product_image_url }}`

URL (Edit Link): [This is Dynamic Link to the Website URL](#)