

# An Introductory Primer to the Google Tag Manager Web Interface

Created by Sanabu Washizuka

v 1.0.0

## Contents

Introduction .....	3
Google Tag Manager Structure.....	3
Setup.....	5
Accounts.....	5
Base Google Tag Manager JavaScript Installation .....	6
Container components .....	7
Workspace.....	7
Publishing.....	8
Tags and Triggers .....	9
Tags .....	9
Variables.....	14
Applying a Variable Through Tag Configuration .....	16
Triggers.....	18
Tag Assembly.....	20
Version control .....	23
Rollback.....	23
Preview .....	25
Contact.....	28

## Introduction

Google Tag Manager is a tag management system that centralizes the creation, editing and deployment of HTML, JavaScript related to web services, providing engineers and administrators who maintain websites an easy way to maintain and dynamically deploy tags.

These tags might include triggers to catch events such as page visits or cart totals from e-commerce websites and send related information to data storage or analytics services.

The expansion of capabilities in a website may necessitate an increase in tags, their complexity, and intractability between elements, making it difficult to update tags when necessary. For example, without a tag management system adding a new product page to an e-commerce store might require the data capturing script tags to be hand-coded again for this new page. Instead of coding out conditions for data capture in JavaScript by hand, website engineers can set conditions through the Google Tag Manager interface. By simply adding a repeatable block of JavaScript into the HTML code of a webpage, website engineers can use Google Tag Manager to automatically create the necessary JavaScript and make it accessible to the webpage HTML file of this new product page while configuring details for the tag's behavior through the Google Tag Manager interface.

## Google Tag Manager Structure

The basic components of Google Tag Manager are tags, triggers and variables.

Tags represent a block of HTML or JavaScript to construct and inject that has a particular function (e.g. report an event to Google Analytics or report a click on an ad to Google Ads).

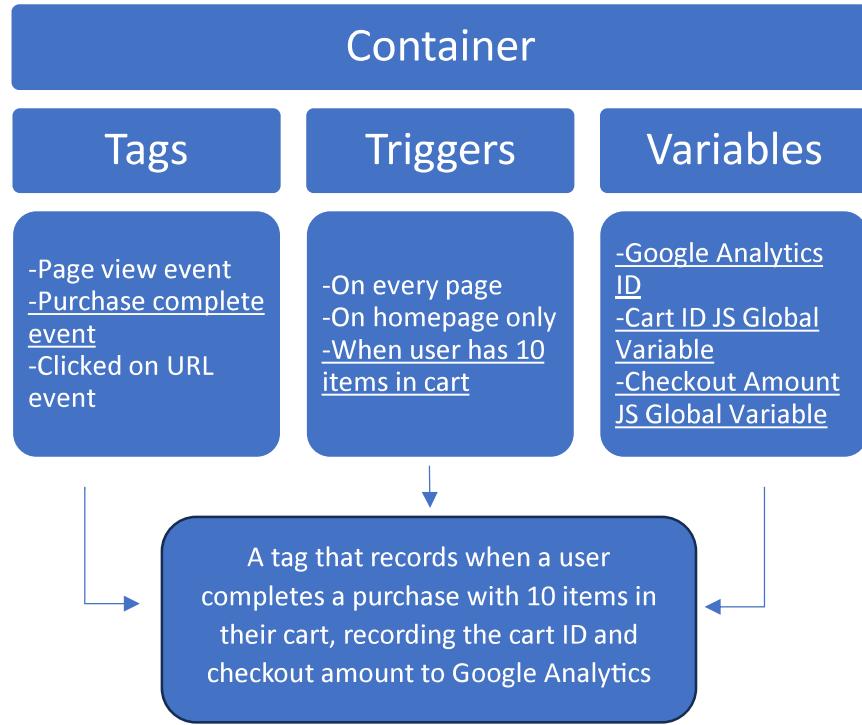
Variables are storage spaces for small pieces of information such as strings or numbers (e.g. input from a user-filled form or a transaction ID).

Triggers define under what conditions a tag will fire (e.g. if the tag is fired at a particular time of day, if the current user is registered to the website).

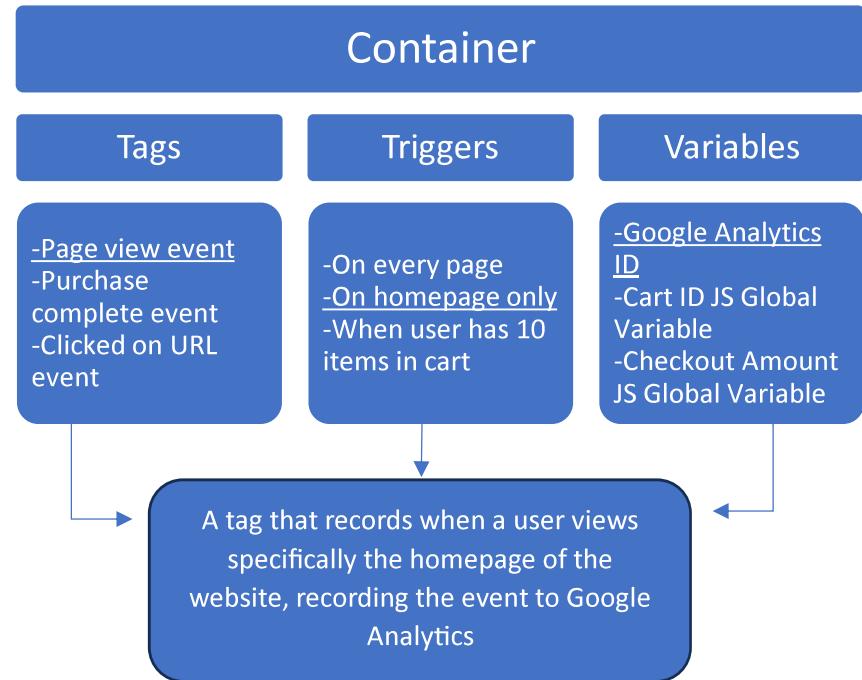
Google Tag Manager makes it easy to rearrange these elements through its web interface, automatically producing and making the necessary JavaScript accessible from the HTML pages sent to users.

The entity that groups all of these elements together and associates them to a website is a container. Each container is usually associated with a single website domain name. Tags have variables and/or triggers applied to them that have been created within the same container. Below are visual examples of the contents of a container and how they might be assembled. Both examples have the same contents within the container, but arrange different elements to configure different tags:

E.g. 1:



E.g. 2:



## Setup

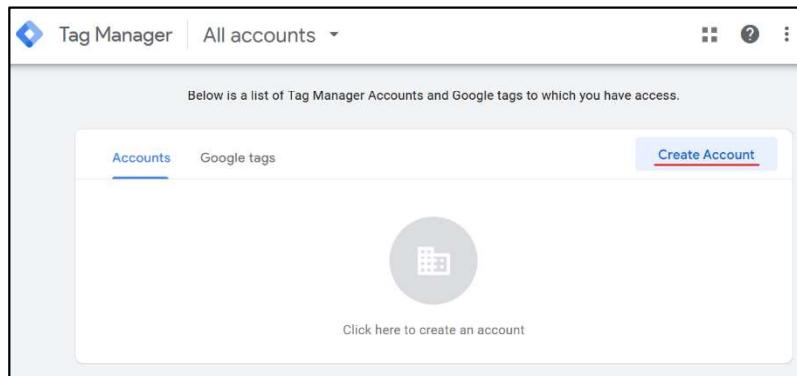
### Accounts

Google Tag Manager can be accessed through the URL below:

tagmanager.google.com

Any Google account can be used to access Google Tag Manager. Log in with a previously existing Google account or create one as prompted if necessary.

Upon your first log in, Google Tag Manager will open to the accounts page. A Google Tag Manager account will be necessary to create and manage containers. To create one, click "Create Account" in the upper right.

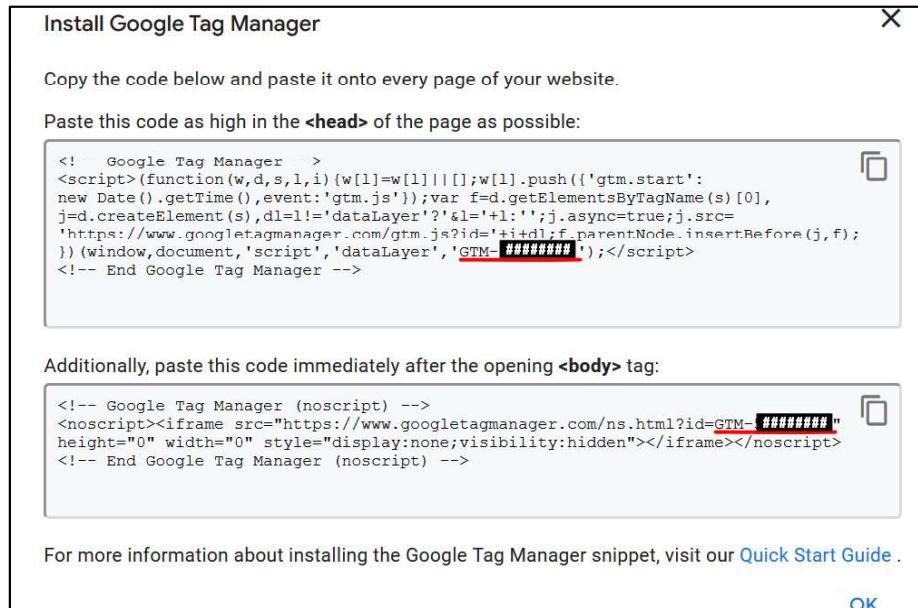


Fill in the form for adding a new account. Container names are conventionally the URL of the website this container will connect to.

For the "Target platform" choose "Web" if connecting to a website.

A screenshot of the 'Add a New Account' dialog box. It has two main sections: 'Account Setup' and 'Container Setup'. In 'Account Setup', there is a 'Account Name' field containing 'e.g. My Company', a 'Country' dropdown set to 'United States', and a checkbox for 'Share data anonymously with Google and others'. In 'Container Setup', there is a 'Container name' field containing 'e.g. www.mysite.com', a 'Target platform' section with a list of options, and a 'Create' button at the bottom. The 'Target platform' section includes icons and descriptions for Web (for desktop and mobile web pages), iOS (for iOS apps), Android (for Android apps), AMP (for Accelerated Mobile Pages), and Server (for server-side instrumentation and measurement).

Creating an account will automatically create a container for the website, generating the Google Tag manager installation JavaScript.



Copy and save elsewhere both JavaScript blocks that appear for later use during the installation of Google Tag Manager into your webpage (This code can be found later in the workspace for the container).

Additionally, take note of the Google Tag Manager ID, in the format of "GTM-#####" where "#" will be alphanumeric characters. This is an identification number unique to the created container. Some tags or other applications may reference this ID.

### Base Google Tag Manager JavaScript Installation

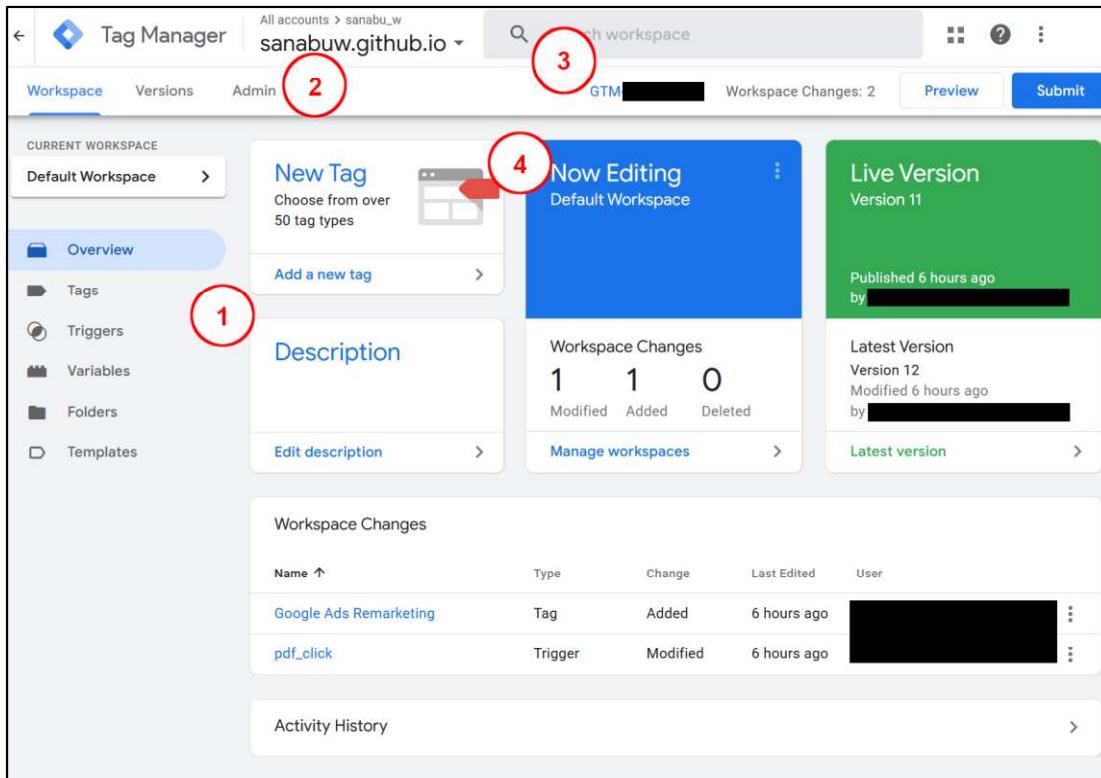
The two code blocks above are the installation script for Google Tag Manager and must be copied into any HTML page that will use Google Tag Manger tags.

Note the instructions that came with the code: the top block is to be placed as high within the <head> </head> tags as possible, and the lower block is to be placed immediately after the opening <body> tag of the HTML files. It is best practice to copy the blocks into all pages that might use any tags in the foreseeable future. The installation itself can be done either through copy/pasting the blocks by hand into each HTML file, or using a Content Management System(e.g. Wordpress) to insert the code across all pages through an entire domain.

## Container components

### Workspace

The workspace is the main interface for creating and configuring tags as well as managing container versions and publishing container configurations to a live website.



To the left (red circle "1" in the image above) is a navigation pane where sections for tags, triggers and variables are accessed. Users will spend most of their time in these three sections of the workspace.

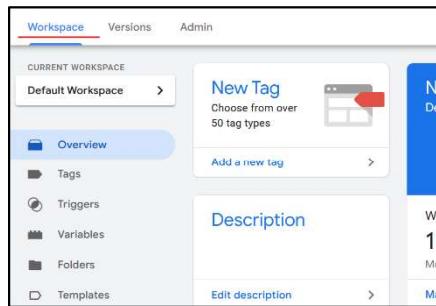
Other sections are accessible through the navigation tabs at the top left of the workspace ("2" above). The workspace itself is accessed on the leftmost tab, "Workspace". Version control is available if any alternate versions of a container are necessary for testing features or if the container must be rolled back to previous versions. Admin controls can be accessed here as well which provide configurations on user accounts and information on the container.

At the top right of the interface ("3" above) is the container's Google Tag Manager code (beginning with "GTM-". Clicking on this will provide the base JavaScript to install Google Tag Manager on a web page. To the right are the "Preview" and "Submit" buttons, enabling the current container within the workspace to be either previewed within the current browser or published live, respectively.

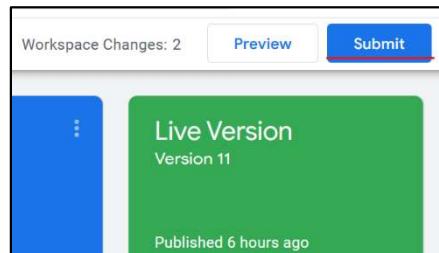
The overview section of the dashboard ("4" above) provides summary statistics of total changes, recent changes, and information on which container version is currently live.

## Publishing

Saved changes to a container do not apply to any live website unless the changes are submitted. In order to submit changes, navigate to the workspace via the tabs in the upper left of the interface.

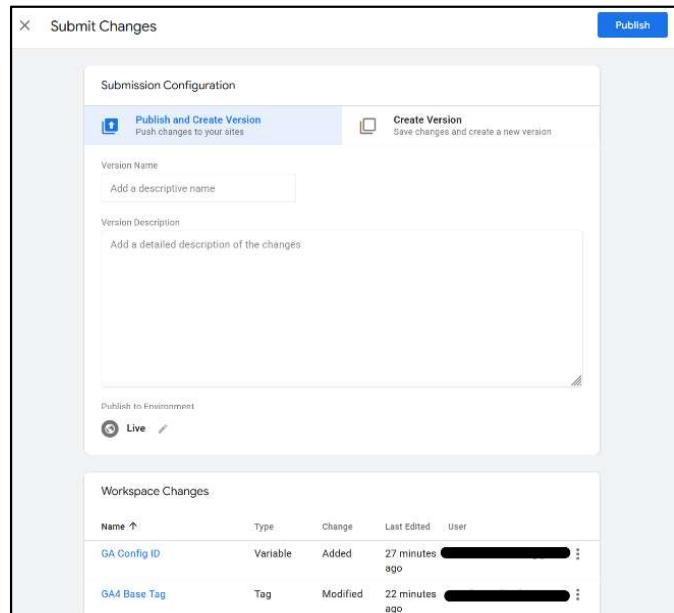


In the upper right of the interface, the "Preview" and "Submit" buttons appear. Click on the "Submit" buttons to begin the process to make changes within the workspace live.



Two main options are provided "Publish and Create Version" and "Create Version". "Create Version" will create a new version from the current container configuration. This is useful when testing out large features that may disrupt work on other parts of the container. "Publish and Create Version" will create a new version in the same way as "Create Version", but will also make the changes live, affecting public users who access the website. A history of changes to the workspace is available for review below the submission configuration window.

Add a version name and version description that represents the main changes within the current version. To finalize and publish the version, click "Publish" in the upper right of the window.



## Tags and Triggers

### Tags

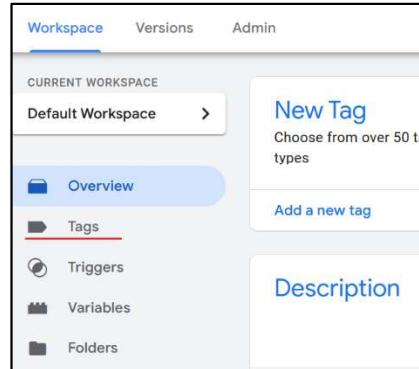
Tags are the core blocks of Google Tag Manager, each designed to perform a specific function. These can be custom-written blocks of JavaScript or HTML, or built-in tags which are solutions created by Google Tag Manager.

#### *Built-in Tags*

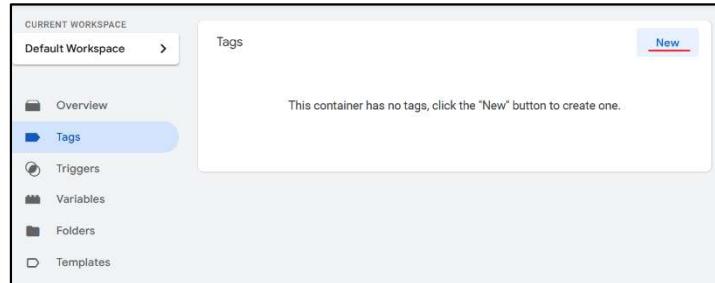
Google Tag Manager provides some tags that are designed to communicate with a service or serve a particular purpose and can be quickly configured through the web interface. For example, the Google Analytics Event tag template provided by Google Tag Manager provides the core block of JavaScript code required for reporting an event (e.g. a page view) back to Google Analytics. From there, users can quickly configure settings like the name of the event that will appear in Google Analytics and any additional key-value parameters to send with the event.

The example below will set up the Google Analytics: GA4 Configuration tag; a built-in tag designed to connect to a Google Analytics account and report common, predefined events back to the account.

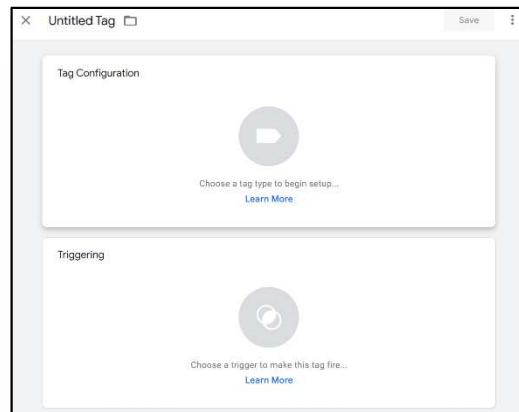
To create a tag, select the Tags section from the left navigation pane.



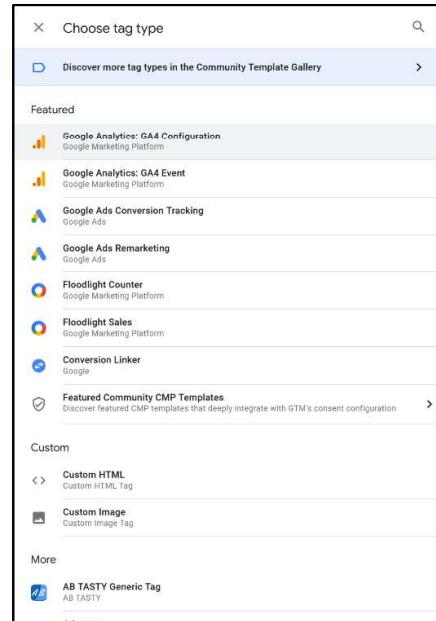
A list of tags within the container is displayed (the example to the right is an empty list). Click "New" in the upper right of the Tags list.



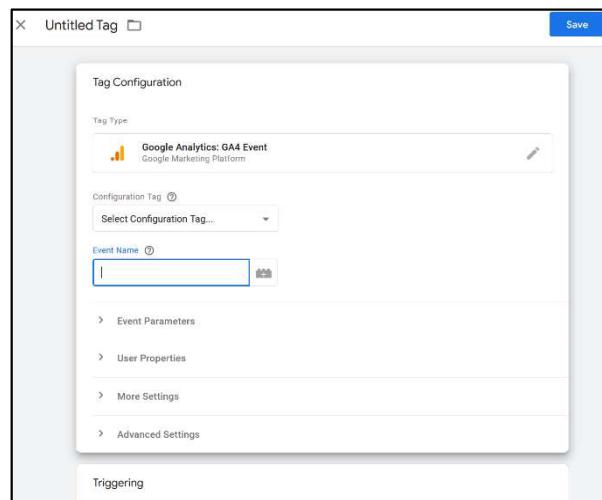
A new pane appears, with prompts to set both the tag configuration (what kind of tag this tag will be) and the trigger conditions for the tag (under what conditions will this tag activate). To set the type of tag, click anywhere in the "Tag Configuration" box.



A list of the built-in tags Google Tag Manager offers appears. Aside from anything under the "Custom" section, almost all these tags will be built-in tags. Selecting a tag type will open detailed configurations for this tag's use. For this example, a Google Analytics GA4 Configuration tag (for recording pre-defined events to a Google Analytics account) will be used.

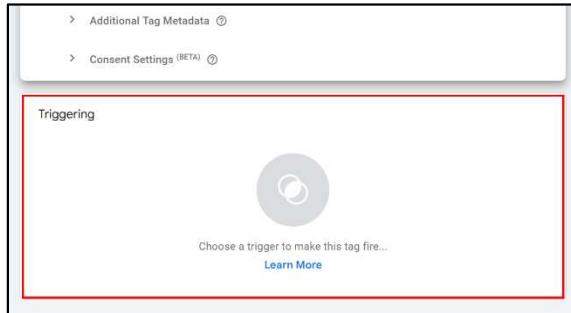


The Tag Configuration pane appears, where detailed settings can be defined for this tag. For the Google Analytics: GA4 Configuration tag, a "Configuration Tag" is necessary. This is the Measurement ID associated with the Google Analytics account (which should be created outside of this tutorial) that will store this data (in the format of "G-#####").



Different settings are available between tag types. Please consult the Google Tag Manager documentation for more detailed information on these tags:

"Your guide to Google Tag Manager"  
<https://support.google.com/tagmanager/answer/12811173>

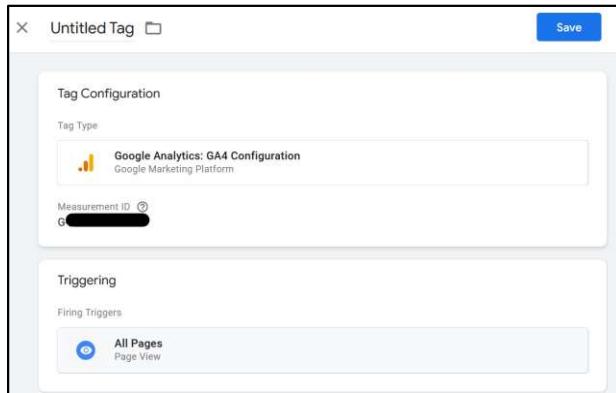


After setting tag configuration, the trigger settings must be made. Scroll down to the "Triggering" box and click anywhere within its borders.

A list of created triggers will appear. Triggers can also be created in the "Triggers" section of the interface (outlined later in this primer). A basic GA4 Configuration tag will normally use "Initialization - All Pages" as its trigger condition. Click on "Initialization - All Pages" (not "All Pages") to set this trigger to fire on any page the user views.

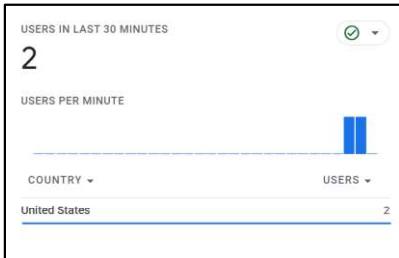
Choose a trigger			
Name ↑	Type	Filter	
All Pages	Page View	--	
Consent Initialization - All Pages	Consent Initialization	--	
<u>Initialization - All Pages</u>	Initialization	--	

The resulting Tag Configuration will now appear as below:



For the GA4 Configuration tag, this will complete the essential setup. Rename the tag by clicking on the upper left text "Untitled Tag" and save the configuration with the upper right button titled "Save". This tutorial will name this tag "GA4 Base Tag".

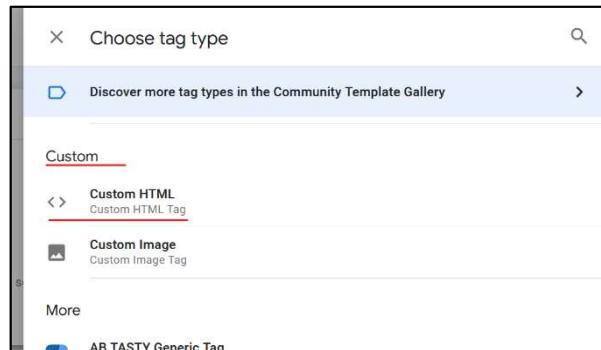
If the base Google Tag Manager tag has been implemented into the HTML, Google Analytics will now receive information on basic events that trigger.



\*This chart is from Google Analytics

### Custom Tags

Custom JavaScript can be implemented as a tag and then managed by Google Tag Manager in terms of when it is triggered, removing the necessity to code in control and conditionals in the JavaScript itself. As an example, a custom browser alert pop-up that reads "Hello, World!" can be created and managed through Google Tag Manager. Custom tags such as these might use the same JavaScript code block that would be used directly in HTML or .js files.



Create a new tag, and when beginning the tag configuration, select "Custom HTML" under the "Custom" section.

The Tag Configuration box appears, including a free-form entry box for any code readable by a browser:

A screenshot of the Google Tag Manager "Tag Configuration" dialog for a tag named "Hello World Tag". The "Tag Type" is set to "Custom HTML". In the "HTML" section, there is a code editor containing the following JavaScript code:

```
1 <script>
2   alert('Hello, World!');
3 </script>
```

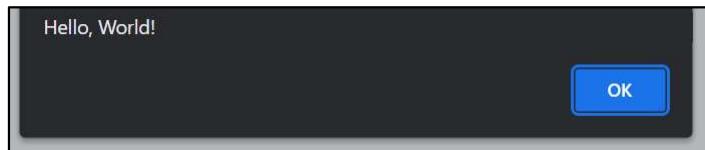
The JavaScript in the image will trigger an in-browser alert pop-up with the text "Hello, World!". Set this to trigger on "All Pages" and rename the tag to "Hello World Tag" to get the resulting configuration below:

The screenshot shows a tag configuration interface. At the top, it says "Hello World Tag". Below that, a green bar says "Added in this workspace" and "Abandon changes". The main area is divided into sections: "Tag Configuration" and "Triggering". In "Tag Configuration", the "Tag Type" is set to "Custom HTML" (with "Custom HTML Tag" as a suggestion). The "HTML" field contains the following code:

```
1 <script>
2   alert('Hello, World!');
3 </script>
```

In the "Triggering" section, under "Firing Triggers", "All Pages" is selected, with "Page View" as a suggestion.

After submitting the changes, users entering the website will see the message below from their browser:

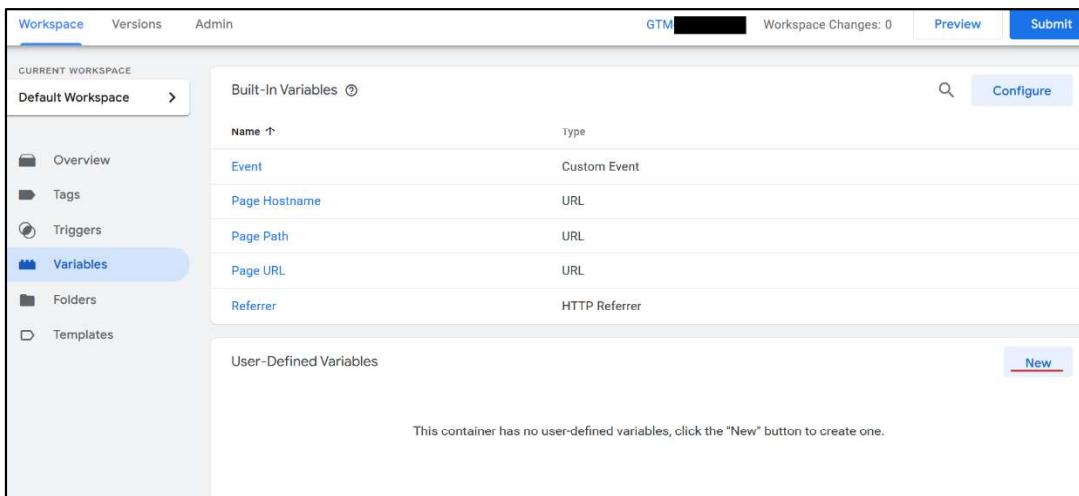


## Variables

Variables, labels that hold a specific piece of data, can be defined for use in Google Tag Manager tags. The data itself can be a constant value defined explicitly through Google Tag Manager or a dynamic value retrieved through JavaScript in the current data layer, session, or browser properties of the user. The examples below demonstrate the creation and implementation of both a constant and dynamic variable.

The constant value variable will be defined first. This variable will contain a generic thank you message that reads "Thank you for visiting this page!" To define a variable, click on "Variables" in the left navigation pane of the workspace. A list of built-in variables and a list of user-defined variables are displayed in the window. The built-in variables are a group of commonly used variables, ready for application into a tag.

To create a custom variable, click on "New" in the upper right of the "User-defined variables" box.

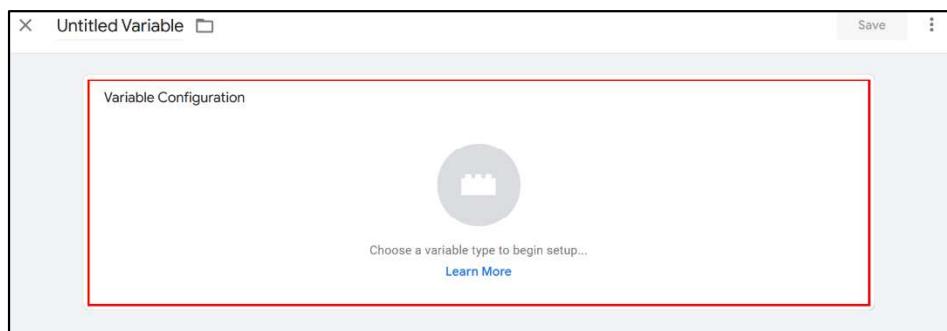


The screenshot shows the Google Tag Manager workspace interface. The top navigation bar includes 'Workspace', 'Versions', 'Admin', 'GTM' (status bar), 'Workspace Changes: 0', 'Preview', and 'Submit'. On the left, a sidebar lists 'CURRENT WORKSPACE' sections: 'Default Workspace' (selected), 'Overview', 'Tags', 'Triggers', 'Variables' (highlighted with a blue box), 'Folders', and 'Templates'. The main content area is titled 'Built-In Variables' with a search icon and 'Configure' button. It displays a table of built-in variables:

Name ↑	Type
Event	Custom Event
Page Hostname	URL
Page Path	URL
Page URL	URL
Referrer	HTTP Referrer

Below this is a section titled 'User-Defined Variables' with a 'New' button. A message at the bottom states: 'This container has no user-defined variables, click the "New" button to create one.'

A new window appears to configure the variable. Click on the "Variable Configuration" box.



The screenshot shows a modal dialog box titled 'Untitled Variable'. The top right corner features a 'Save' button and a more options menu. The main area is labeled 'Variable Configuration' and contains a large red-bordered box. Inside this box is a circular icon with a gear-like symbol and the text 'Choose a variable type to begin setup...'. Below the icon is a 'Learn More' link. The entire dialog box is set against a light gray background.

A list of variable types appears. Each variable type has different configurations possible. To enter a custom, predefined value, select the "Constant" type under "Utilities".

The screenshot shows a configuration window for "Utilities". It lists three options:

- Element Visibility**: The value is set based on the visible state of the specified DOM element.
- Constant**: The value is set to the string you provide. This option is currently selected, indicated by a red underline.
- Custom Event**: The value is set to "eventNameXYZ" when the following code on your website is executed:  
dataLayer.push({event: 'eventNameXYZ'});

In the next window that appears, under "value", type in "Thank you very much!". Rename the variable "thankyou\_msg" and save the configuration.

The screenshot shows the "Variable Configuration" window for a variable named "thankyou\_msg". The configuration details are as follows:

- Variable Type**: Constant
- Value**: Thank you very much!

A blue "Save" button is located in the top right corner.

Next, the dynamic value variable will be defined that contains the current time within the user's browser's environment. A JavaScript function will be required to retrieve this from the user's browser. Create another new variable with the type "Custom JavaScript" found under "Page Variables":

The screenshot shows the "Page Variables" configuration window. It lists four variable types:

- 1st Party Cookie**: The value is set to the first value of the 1st party cookie with the matching name.
- Custom JavaScript**: This variable uses the provided JavaScript function to calculate its value in the browser. Each time this variable is used, the function will be executed and its return value will be used. This option is currently selected, indicated by a red underline.
- Data Layer Variable**
- JavaScript Variable**: The value is set to that of the global JavaScript variable you specify.

A JavaScript editor appears within the "Variable Configuration" box. A JavaScript function that returns a value can be entered in this box. In this case, a function to retrieve a timestamp of the current time according to the user's browser will be used.

Rename the variable to "get\_date" and save the configuration.

```
function () {
  return new Date();
}
```

These variables will be used later in the "Tag Assembly" section of this primer.

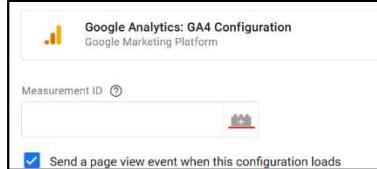
### Applying a Variable Through Tag Configuration

Below is an example of using a variable as a tag configuration setting. Suppose for the GA4 Basic Configuration tag's Configuration ID, we wish to use a variable instead of typing out the ID literally for this tag, in case the ID might be used elsewhere within Google Tag Manager.

Open the configuration window for the Google Analytics: GA4 Configuration Tag by navigating to the Tags section, selecting the tag type "Google Analytics GA4 Configuration" and clicking on the tag configuration box.



Delete what is currently in the Measurement ID box and click on the building block icon to the right of the box.



A window with a list of variables to choose from is displayed. A new variable can be created from this window as well. Create a new variable by clicking on the "+" icon in the upper right.

Create a variable of type constant and enter in the string for a Google Analytics ID under the "Value" field. Rename the variable to "GA Config ID" and save.

Google Tag Manager will now return to the GA4 Configuration tag's configuration window. The measurement ID is now filled in with the variable name for the Google Analytics Configuration ID. The double curved brackets tell Google Tag Manager that this is a variable.

The tag will now use the value stored within the "GA Config ID" variable. If the ID is used across multiple tags and must be changed for any reason, this change can easily be done for all instances the ID is used. It is now only necessary to change the value for the variable itself as opposed to configuring each, individual tag that uses this value.

## Triggers

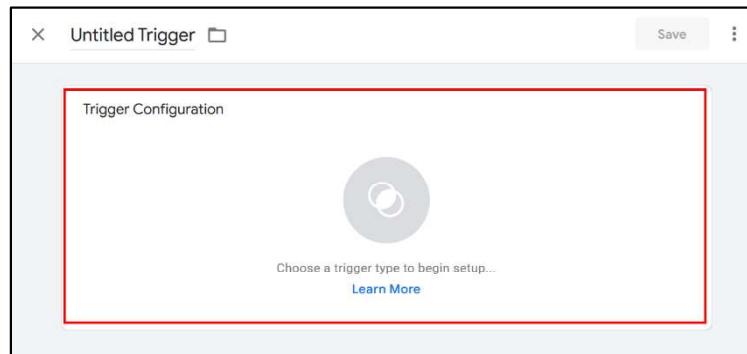
Triggers define under what conditions a tag will fire. Although they can be applied to tags, like variables, they are separate entities from an individual tag within a container. A single trigger configured in a particular way may be applied to different tags.

The example below creates a trigger that fires a tag only when a link to a pdf file is clicked on. This trigger will be applied to a tag in the next section.

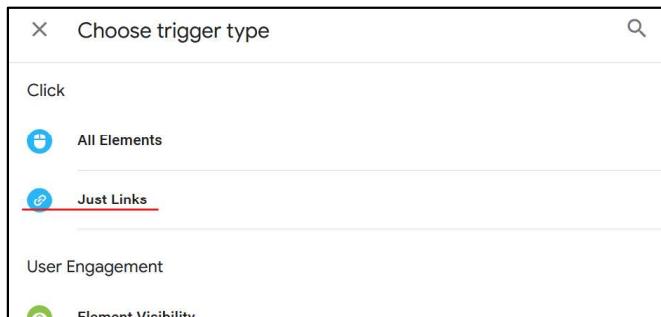
Navigate to the Triggers section from the left navigation panel. Click "New" on the upper right of the triggers list.



In the new window that appears, click on the "Trigger Configuration" box.



A list of trigger types appears. For this example, use the "Just Links" type under the "Click" category.



The trigger configuration provides options for what condition this trigger should fire. Several options are available including looking at page URLs, hostnames, or other custom-made variables. To set the trigger to fire when viewing any pdf file on the site, select "Some Link Clicks" under "This trigger fires on". Set the variable to reference to "Click URL", the search condition "contains" and the search value ".pdf" like below:

The screenshot shows the 'Trigger Configuration' dialog box for a trigger named 'pdf\_click'. At the top right is a blue 'Save' button. Below it is a dark header bar with the text 'Modified in this workspace' and buttons for 'View changes' and 'Abandon changes'. The main area is titled 'Trigger Configuration' and contains a 'Trigger Type' section with a radio button selected for 'Click - Just Links'. Below this are two unchecked checkboxes: 'Wait for Tags' and 'Check Validation'. Under 'This trigger fires on', a radio button is selected for 'Some Link Clicks'. At the bottom, there's a search bar with the placeholder 'Fire this trigger when an Event occurs and all of these conditions are true', containing the text 'Click URL contains .pdf'.

Rename the trigger to pdf\_click and save the trigger.

The trigger is now included in the triggers list and is ready to be used in tags.

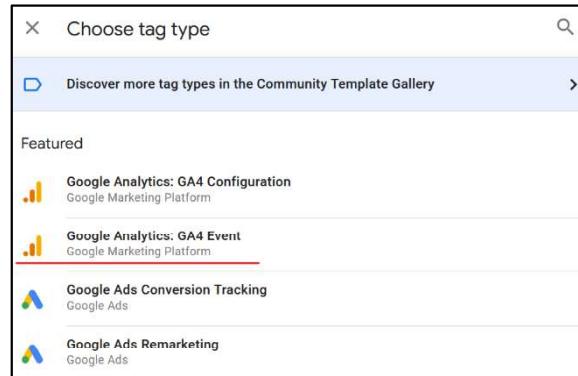
Triggers					
<input type="checkbox"/>	Name <span>↑</span>	Event Type	Filter	Tags	Last Edited
<input type="checkbox"/>	pdf_click	Just Links	Click URL contains .pdf	2	a few seconds ago

## Tag Assembly

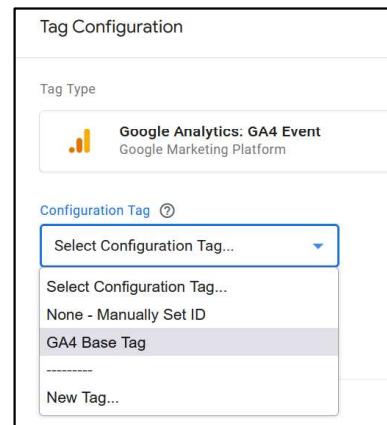
Different variables and triggers can be assigned easily and dynamically to different tags with Google Tag Manager. The assignment of variables and triggers to a tag is done through tag configuration. We will now create two new events: one to send information to Google Analytics that a user viewed a pdf on the site and a second to send a message to the user through the browser that includes a thank you message and information on when they viewed a pdf.

The Google Analytics event tag for viewing a pdf will use the pdf\_click trigger to send an event to Google Analytics when any user clicks on a link to a pdf file on the website.

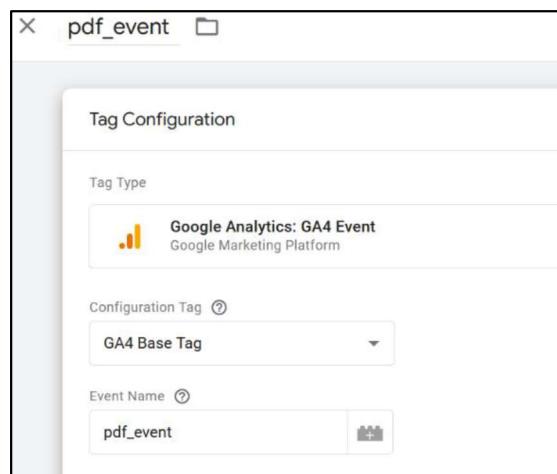
Create a new tag with tag type GA4 Event:



To set the Configuration Tag, because the Google Analytics: GA4 Configuration tag has been set, an option to use the same ID is now available as "GA4 Base Tag". Select this option for the Configuration Tag.



"Event Name" will be the name that represents this event within Google Analytics. This example will name this event "pdf\_event".



In the triggering configuration, select the previously made "pdf\_click" trigger.

The image shows two screenshots from Google Tag Manager. The top screenshot is titled 'Choose a trigger' and lists several triggers: 'All Pages' (Page View), 'Consent Initialization - All Pages' (Consent Initialization), 'Initialization - All Pages' (Initialization), and 'pdf\_click' (Just Links). The 'pdf\_click' trigger is selected and highlighted with a red border. The bottom screenshot shows the configuration for a tag named 'pdf\_event'. Under 'Tag Configuration', it is set to 'Google Analytics: GA4 Event' (Tag Type) and 'GA4 Base Tag' (Configuration Tag). The 'Event Name' is set to 'pdf\_event'. Under 'Triggering', it is configured to fire on 'pdf\_click' (Just Links). Both screenshots have a search bar at the top right and a '+' button for creating new triggers.

The image below shows the resulting configuration.  
Save the trigger.

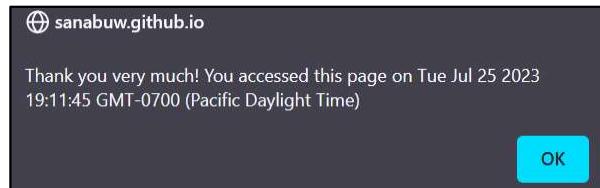
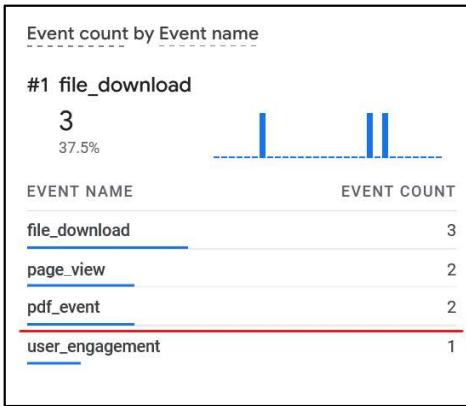
For the tag that will send a message via the browser to a user, a custom JavaScript tag will be used. Variables created within Google Tag Manager can also be used in custom JavaScript tags. This example will use the `thankyou_msg` message variable and the `get_date` variable, which retrieves a timestamp of the current time from the user's browser. Use the Custom HTML tag type when creating this tag:

The image shows the 'Tag Configuration' screen for a 'Custom HTML' tag. Under 'Tag Type', it is selected as 'Custom HTML' (Custom HTML Tag). In the 'HTML' section, the following code is entered:

```
1 <script>
2   alert('{{thankyou_msg}} You accessed this page on {{get_date}});
3 </script>
```

This tag is intended to trigger under the same condition the GA4 event tag, when any pdf link is clicked on. The trigger to apply will be "pdf\_link".

Save the trigger and publish the container. Clicking on a pdf will now send pdf\_event events to Google Analytics and will display the message to the user, thanking them and providing a timestamp of their access through a browser alert window.



\*Message from the user's browser

\*From Google Analytics

## Version control

Google Tag Manager records all changes made in a container. Every time a container is published, a new version of the container is created. Different branches can also be created in case different Google Tag Manager users wish to work on and test different functions.

## Rollback

If an issue is discovered with the latest or live version of a container, the container can be rolled back to an older version, changing all current configurations to that of the previous version. Versions can be reviewed through the "Versions" tab at the top left of the Google Tag Manager interface.

The screenshot shows the 'Versions' tab in the Google Tag Manager interface. At the top, it displays 'sanabuw.github.io' and 'GTM-[REDACTED]'. It shows 'Version 9 is Live' was published on Jul 25, 2023, by [REDACTED]. Below this, there are counts for Tags (5), Trigger (1), and Variables (2). The main area is titled 'Versions' and contains a table:

Version ID	Status	Name	Created	Published	Published By
9	Live, Latest	erroneous version	07/25/2023	07/25/2023	[REDACTED]
8			07/25/2023	07/25/2023	[REDACTED]
7			07/25/2023	07/25/2023	[REDACTED]
6		pause Hello World	07/25/2023	07/25/2023	[REDACTED]
5		Hello world added	07/25/2023	07/25/2023	[REDACTED]
4			07/25/2023	07/25/2023	[REDACTED]
3			07/25/2023	07/25/2023	[REDACTED]
2		GA4 Configuration added	07/25/2023	07/25/2023	[REDACTED]
1		Empty Container	07/25/2023	07/25/2023	[REDACTED]

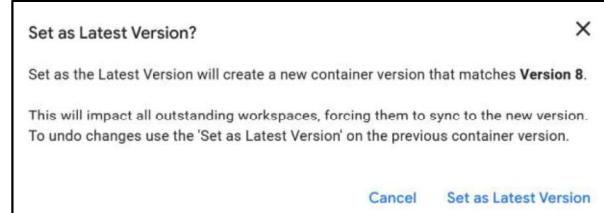
The example above shows a version with version ID 9 that had issues on release. We wish to revert to the last successful version, Version ID 8.

Click on the ellipses menu icon to the right of the version to revert back to and select "Set as Latest Version".

The screenshot shows the 'Versions' tab in the Google Tag Manager interface. The table from the previous screenshot is shown again. A context menu is open over the row for Version 8, which is highlighted with a red underline. The menu options are:

- Preview
- Share Preview
- Publish
- Rename
- Set as Latest Version** (this option is underlined)
- Delete
- Export
- Edit Notes

A window appears prompting to confirm the change. Click "Set as Latest Version" to commit to this change and make the desired version the current newest version.



After setting a version to the latest version, the versions list has a new entry, Version ID 10 with the name "Restored\_8", indicating a new version has been made from version 8's configuration.

The screenshot shows a web interface for managing website versions. At the top, it displays "sanabuw.github.io" and "Version 9 is Live". It also shows statistics: 5 Tags, 1 Trigger, and 2 Variables. Below this, it says "Published Jul 25, 2023 by [REDACTED]". The main area is titled "Versions" and contains a table:

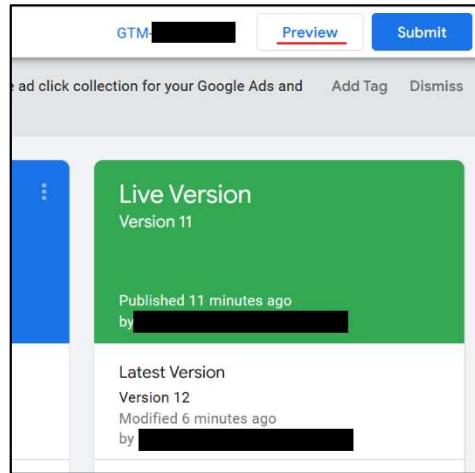
Version ID	Status	Name	Created	Published	Published By
10	Latest	Restored_8	07/26/2023		[REDACTED]
9	Live	erroneous version	07/25/2023	07/25/2023	[REDACTED]
8		successful publish	07/25/2023	07/25/2023	[REDACTED]
7			07/25/2023	07/25/2023	[REDACTED]
6		pause Hello World	07/25/2023	07/25/2023	[REDACTED]
5		Hello world added	07/25/2023	07/25/2023	[REDACTED]

Note the problematic version, Version 9, is still the live version. To apply the newest, reverted version to the live website, submit and publish the current version as normal.

## Preview

The preview mode is used to debug behavior of a version before it goes live. Preview mode provides real time insights on what tags are or are not activated, including what conditions triggered or failed as well as behavior following the firing of a tag.

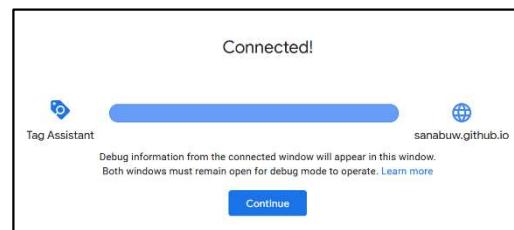
In order to preview a version, click the "Preview" button on the upper right of the interface.



Two new pages load within new tabs in the user's browser. One page is the preview version of the container's website that has all the current container versions' configurations applied to it. This version of the page is live only for the current browser used to activate Google Tag Manager's preview mode.

The other page is the Tag Assistant, Google Tag Manager's debugging interface. A window overlays on top of this page first to provide information on connection status to the website.

Clicking Continue will move the user to Tag Assistant.



To the left ("1" in the image above) is a navigation pane where either a summary of all events can be or a specific event can be reviewed. Executing an action (e.g. clicking on a URL) will populate this list with that action.

To the right ("2" above) is the output for the summary or action selected. At the top is a navigation pane where information on tags, variables, the data layer and other information helpful for debugging can be navigated to.

The tags section provides a list of tags and information on which tags have and have not fired. Continuing with examples created in this primer, we will attempt to trigger the pdf\_event tag and thankyou\_event tag by clicking on a pdf link.

As an example, there are reports that the event pdf\_event is not being recorded and the thank you message for viewing the pdf is not appearing to users. This example will use Tag Assistant to gain insight on what the issue might be.

After a pdf link is clicked in the preview window of the website with the current, working container version, we see in the history the "Link Click" action, representing the user's action clicking on a link.

Selecting this will show what tags did or didn't trigger on this action.

Event: Link Click

API Call

```
dataLayer.push({event: "gtm.linkClick", ...})
```

Output of GTM- [REDACTED] ② Previewing: env-4 >

Tags Variables Data Layer Consent Errors

Tags Fired None

Tags Not Fired

GA4 Base Tag Google Analytics: GA4 Configuration	Hello World Tag Custom HTML - Paused	pdf_event Google Analytics: GA4 Event	thankyou_event Custom HTML
-----------------------------------------------------	-----------------------------------------	------------------------------------------	-------------------------------

The tags pdf\_event and thankyou\_event are expected to have fired, but they are under the "Tags Not Fired" section of the output report. Click on pdf\_event to review details about what caused the tag not to fire.

Under "Firing Triggers" the filter "Click URL" has a red cross next to it, indicating that this condition did not pass. Upon further inspection to the right along that row, we see the value to compare on was ".pdg", implying the incorrect string was input during tag configuration. Reviewing the tag configuration for the trigger pdf\_click and replacing the value to check to ".pdf" will resolve this issue.

6 Link Click > pdf\_event ✘ Not Fired

Tag Details Display Variables as  Names  Values

Name	Value
Type	Google Analytics: GA4 Event
Firing Status	Not fired
Send Ecommerce data	false
Event Name	"pdf_event"
Configuration Tag	" GA Config ID "

Firing Triggers

✘ pdf_click
Filters
✘ Click URL contains .pdg
✓ _event equals gtm.linkClick
✓ _triggers matches RegEx (^\$ ((^.)136187396_8(\$ )))

Blocking Triggers  
No blocking triggers

## Contact

If you have any questions, please kindly reach out to me with any questions at the following methods below:

E-mail: [sanabuwashizuka@gmail.com](mailto:sanabuwashizuka@gmail.com)

LinkedIn: [www.linkedin.com/in/sanabuwashizuka](https://www.linkedin.com/in/sanabuwashizuka)

Portfolio: <https://sanabuw.github.io/>