


# Google Analytics & GTM

## Google Analytics是什么

 analytics.js 库（也称为“Google Analytics（分析）代码”）是一种用于衡量用户与您网站的互动情况的 JavaScript 库。本文介绍了如何将 Google Analytics（分析）代码添加到您的网站中。（**这个我们通常叫GA 文档:** <https://developers.google.com/analytics/devguides/collection/analyticsjs?hl=zh-cn>）（**2023年7月1号正式废弃**）

Google Analytics（分析）4 是一项分析服务，让您可以使用自定义报告来衡量网站和移动应用的流量和互动度。本文档提供了适用于开发者群体的实现说明和参考资料。（**这个我们叫它GA4文档:** <https://support.google.com/analytics/answer/9306384?hl=zh-Hans>）

全局网站代码 (gtag.js) 是一种 JavaScript 代码植入框架兼 API，供您用来向 Google Analytics（分析）、Google Ads 和 Google Marketing Platform 发送事件数据。本页内容介绍如何使用 gtag.js 来配置要部署 Google Analytics（分析）的网站。如需通用的（即并非仅适用于某款产品的）gtag.js 说明文档，请参阅 [gtag.js 开发者指南](#)。（**这个我们叫它 Gtag**）

## GA

- **目前chime大部分项目的埋点都是基于GA实现的**
- GA是一个埋点工具, 早期还没有GA4
- ID都是以UA开头的, 例如: UA-xxxxx
- 使用GA自己的API控制埋点过程, 不过现在可以迁移到Gtag API ( <https://developers.google.com/analytics/devguides/migration/ua/analyticsjs-to-gtagjs> )
- **GA目前已经兼容了GTag API**

## GA4

- 2020年10月14日google推出了的GA4, 现在默认创建的埋点都是用的这个工具, 但是你也可以手动创建基于GA的埋点
- 和GA的不同: [https://support.google.com/analytics/answer/10315383?hl=zh-Hans&ref\\_topic=9303319](https://support.google.com/analytics/answer/10315383?hl=zh-Hans&ref_topic=9303319)
- ID都是以G开头的, 例如: G-xxxxx
- 使用Gtag的API控制埋点过程

# GTag

- 这个就是一套API, google的一些产品( Google Ads、 Campaign Manager、 Display & Video 360、 Search Ads 360 和 Google Analytics 4 )都是用的这套API
- GA和Gtag API的区别

| API       | GA  | Gtag   |
|-----------|---|--|
| 初始化       | <pre>JavaScript  1 ga('create',     'GA_MEASUREMENT_ID', 'auto');</pre>   | <pre>JavaScript  1 window.dataLayer =   window.dataLayer    []; 2 function gtag()   {dataLayer.push(arguments);} 3 gtag('js', new Date()); 4 gtag('config',     'GA_MEASUREMENT_ID');</pre>  |
| Page view | <pre>JavaScript  1 ga('send', 'pageview');</pre>  | <pre>JavaScript  1 gtag('event', 'page_view', {   'send_to': 'GA_MEASUREMENT_ID' }); 2 gtag('event', 'page_view', {   'send_to':   ['GA_MEASUREMENT_ID1',   'GA_MEASUREMENT_ID2']});</pre>   |
| 事件        | <ul style="list-style-type: none"><li>• 事件基于category, action, label, value</li><li>• category, action必填, label, value选填</li></ul> <pre>JavaScript  1 ga('send', 'event',   [eventCategory],   [eventAction], [eventLabel],   [eventValue],   [fieldsObject]);</pre> | <ul style="list-style-type: none"><li>• 事件基于一个对象</li></ul> <pre>JavaScript  1 gtag('event', eventName,   eventParameters); 2 // 老的GA要这样兼容 3 gtag('event', &lt;eventName&gt;, { 4   'event_category':   &lt;category&gt;, 5   'event_label': &lt;label&gt;, 6   'value': &lt;value&gt;, 7   'send_to':   ['GA_MEASUREMENT_ID1',   'GA_MEASUREMENT_ID2'] 8 });</pre> |

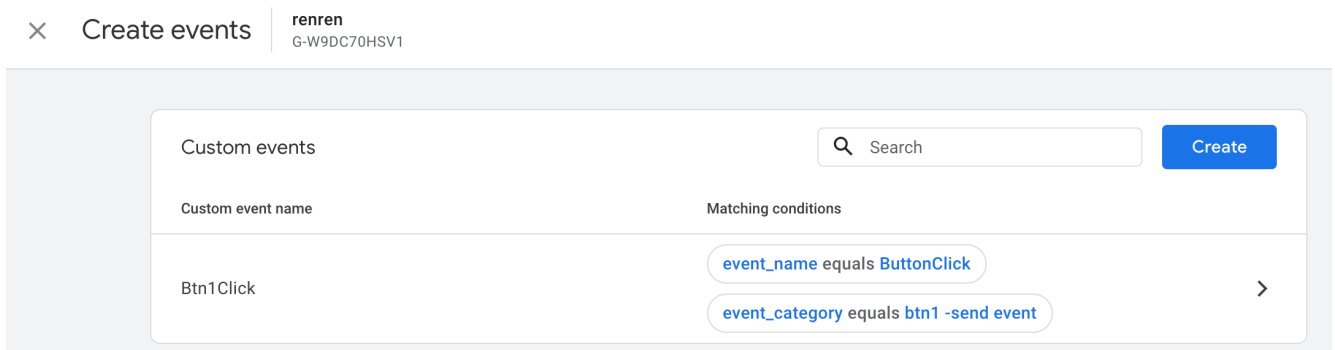
|                  |   |  |
|------------------|---|--|
| 自定义<br>维度和<br>指标 | <pre>JavaScript 1 ga('send', 'hitType', {   'dimension&lt;Index&gt;':   'dimension_value'});</pre>  | <pre>JavaScript 1 gtag('event',   'any_event_name',   {'dimension_name':   'dimension_value'});</pre>  |
| 用户计<br>时         | <pre>JavaScript 1 ga('send', 'timing',   'timingCategory',   'timingVar', timingValue,   'timingLabel');</pre>  | <pre>JavaScript 1 gtag('event',   'timing_complete', { 2   'name': 'timingVar', 3   'value': timingValue, 4   'event_category':   'timingCategory', 5   'event_label':   'timingLabel' 6 });</pre> |
| 异常               | <pre>JavaScript 1 ga('send', 'exception', { 2   'exDescription':   'error_message', 3   'exFatal': false // set   to true if the exception is   fatal 4 });</pre>   | <pre>JavaScript 1 gtag('event', 'exception', { 2   'description':   'error_message', 3   'fatal': false // set to   true if the exception is fatal 4 });</pre>                                     |
| 字段对<br>应         | <a href="https://developers.google.com/analytics/devguides/migration/ua/analyticsjs-to-gtagjs#map_analyticsjs_fields_to_gtagjs_parameters">https://developers.google.com/analytics/devguides/migration/ua/analyticsjs-to-gtagjs#map_analyticsjs_fields_to_gtagjs_parameters</a> |  |

**!** 事件埋点是埋点统计的核心, **category, action, label**应该怎么定义, 这取决于你要怎么统计这些数据, 有关详细内容可以好好看下 <https://support.google.com/analytics/answer/1033068>

## GA4相关

### • Events, 参考文档

- 可以通过配置的方式创建或者修改事件
- 定制事件的目的:
  - 在不同的媒体资源和数据流中使用相同的事件名称，从而简化报告
  - 修正事件名称、条件或参数中的拼写错误导致的衡量错误
  - 根据现有事件创建新事件，并在特定条件下触发该事件
  - 缩小事件范围并将其标记为转化
- 定制事件是有限制的

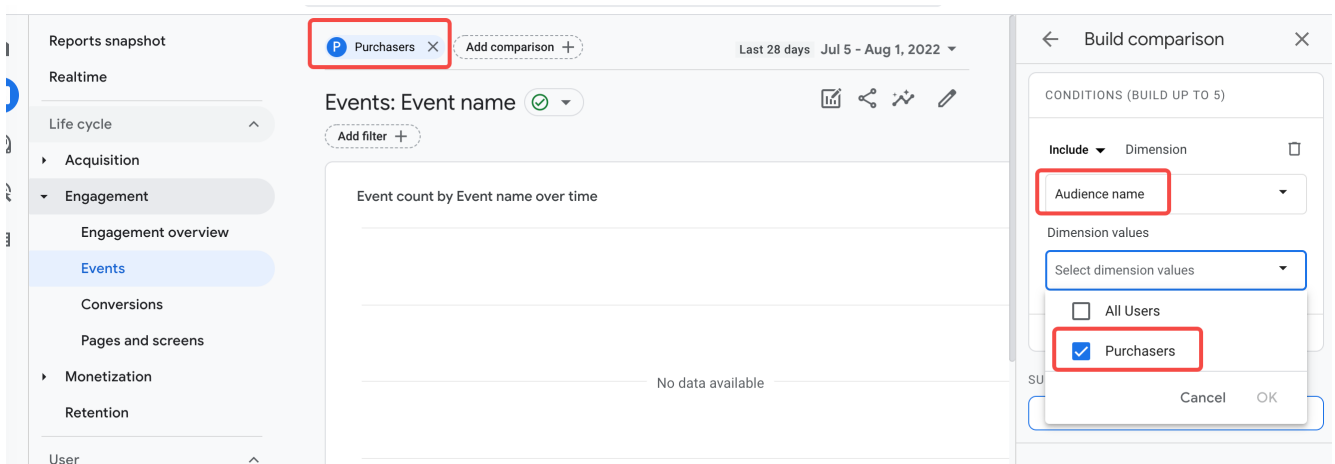


### • Conversions (转化), 参数文档

- 可以和Google Ads结合起来使用

### • Audience (受众), 参考文档

- Audience 有点像之前的视图(filter view), 但是它更侧重于数据的分类和比较
- 默认有Purchasers和All Users两个Audience, 可以在报表里面使用



### • 维度和指标, 参考文档

- 您应该先充分利用通过[自动收集的事件](#)、[增强型衡量事件](#)以及[推荐事件](#)收集的维度和指标，然后再设置自定义维度和指标。

- 维度分为**用户级**和**事件级**, 避免创建**高基数**的自定义维度
- 预定义的用户维度, 参考[文档](#)
- 必须达到 [Google Analytics \(分析\) 数据阈值](#)才能查看受众特征和兴趣的数据
- 指标可以理解为对数据的一种聚合处理逻辑
- [数据抽样 & 非抽样数据报告 & 未抽样数据探索](#)

## 其他相关产品

### Google Ads

Google ads使用了Gtag API, ID都是以AW开头的, 例如: AW-xxxxx, 调用方式如下

#### JavaScript

```
1 // 初始化
2 gtag('config', 'AW-1004917963');
3 // 触发转化
4 gtag('event', 'conversion', {
5     'send_to': 'AW-1004917963/e2YNCPCgpIUBEMupl98D'
6 });
```

## Google Tag Manager是什么



Google Tag Manager 是一个标签管理系统 (TMS), 可让您快速轻松地更新衡量代码和相关代码片段, 这些代码片段统称为您网站或移动应用程序上的标签。将一小段跟踪代码管理器代码添加到您的项目后, 您可以从基于 Web 的用户界面安全轻松地部署分析和测量代码配置。

## 初始化

GTM-KVVFJ33 Workspa

Install Google Tag Manager

Copy the code below and paste it onto every page of your website.

Paste this code as high in the <head> of the page as possible:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i)(w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-KVVFJ33');
<!-- End Google Tag Manager -->
```

## 代码(Tag)

- 要执行的动作, 至少要有一个触发器(Trigger)结合使用(或者设置Fire a tag after xxxTag fires)

Tag Sequencing ⓘ

Fire a tag before **All Measure Ids Config** fires

Fire a tag after **All Measure Ids Config** fires

Cleanup Tag

PageView >

Don't fire **PageView** if **All Measure Ids Config** fails or is paused

- 可以自己指定Tag Type, 表示要干什么事, 这里有和各个平台有关的Tag(eg: GA), 或者也可以自定义(Custom HTML)

Choose tag type

Discover more tag types in the Community Template Gallery

- Google Analytics: GA4 Event  
Google Marketing Platform
- Google Ads Conversion Tracking  
Google Ads
- Google Ads Remarketing  
Google Ads
- Floodlight Counter  
Google Marketing Platform
- Floodlight Sales  
Google Marketing Platform
- Conversion Linker  
Google
- Google Optimize  
Google Marketing Platform
- Google Surveys Website Satisfaction  
Google Marketing Platform
- Featured Community CMP Templates  
Discover featured CMP templates that deeply integrate with GTM's consent configuration

Custom

- Custom HTML  
Custom HTML Tag
- Custom Image  
Custom Image Tag

## 触发器(Trigger)

- 指定一个Tag的执行时机, 必须要结合一个Tag使用。
- 下面是内置的触发器

Choose trigger type

Page View

- Consent Initialization
- Initialization
- Page View
- DOM Ready

- Window Loaded

---

- Click
  - All Elements
  - Just Links

---

- User Engagement
  - Element Visibility
  - Form Submission
  - Scroll Depth

- 每个触发器都可以设置一些触发条件, 例如:

This trigger fires on

- All Page Views  Some Page Views

Fire this trigger when an Event occurs and all of these conditions are true

Page URL contains /demo-page - +

- 常用的触发器触发时机, 参考[文档](#)
  - Click - All Elements 页面元素点击(这里指的是除了以外的元素)
  - Click - Just Links 页面链接点击(这里指的是元素)
  - Element Visibility 如果要监听的元素是临时插入的, 需要把trigger里面的Observe DOM changes勾上
  - Custom Event 通过代码来触发

Trigger Type

Custom Event

Event name

custom-event1

Use regex matching

This trigger fires on

- All Custom Events  Some Custom Events

## JavaScript



```
1 window.siteDataLayer.push({
2   event: 'custom-event1', // 上图里面的Event name
3   'custom-event-arg1': 'hehe'
4 });
```

## 变量(Variables)





× Choose variable type

Discover more variable types in the Community Template Gallery




Navigation

-  **HTTP Referrer**  
The value is set to the HTTP referrer.
-  **URL**

Page Variables

-  **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 time this variable is used, the function will be executed and its return value will be used.
-  **Data Layer Variable**
-  **JavaScript Variable**  
The value is set to that of the global JavaScript variable you specify.

Page Elements

-  **Auto-Event Variable**
-  **DOM Element**  
The value is set to the text of the DOM element or the value of the specified DOM attribute.
-  **Element Visibility**  
The value is set based on the visible state of the specified DOM element.

## 常用的Variables如下

- Custom JavaScript: 一个匿名方法, 返回值就是这个Variable的值

Custom JavaScript 

```
1 function(){
2   return {
3     'All Measure Ids': {{All Measure Ids}},
4     'custom-event-arg1': {{custom-event-arg1}},
5     'form element': {{form element}},
6     'userId': {{userId}},
7     'Click Element': {{Click Element}},
8     'Click ID': {{Click ID}}
9   }
10 }
```



- Data Layer Variable: 专门用于接收代码层面的传参的Variable

custom-event-arg1 Data Layer Variable

#### JavaScript

```
1 window.siteDataLayer.push({
2   event: 'custom-event1',
3   'custom-event-arg1': 'hehe' // 代码里面的custom-event-arg1属性的值"hehe", 是
   被上图的Data Layer Variable接收的
4 });
```

- JavaScript Variable: 用于接收Javascript变量的Variable

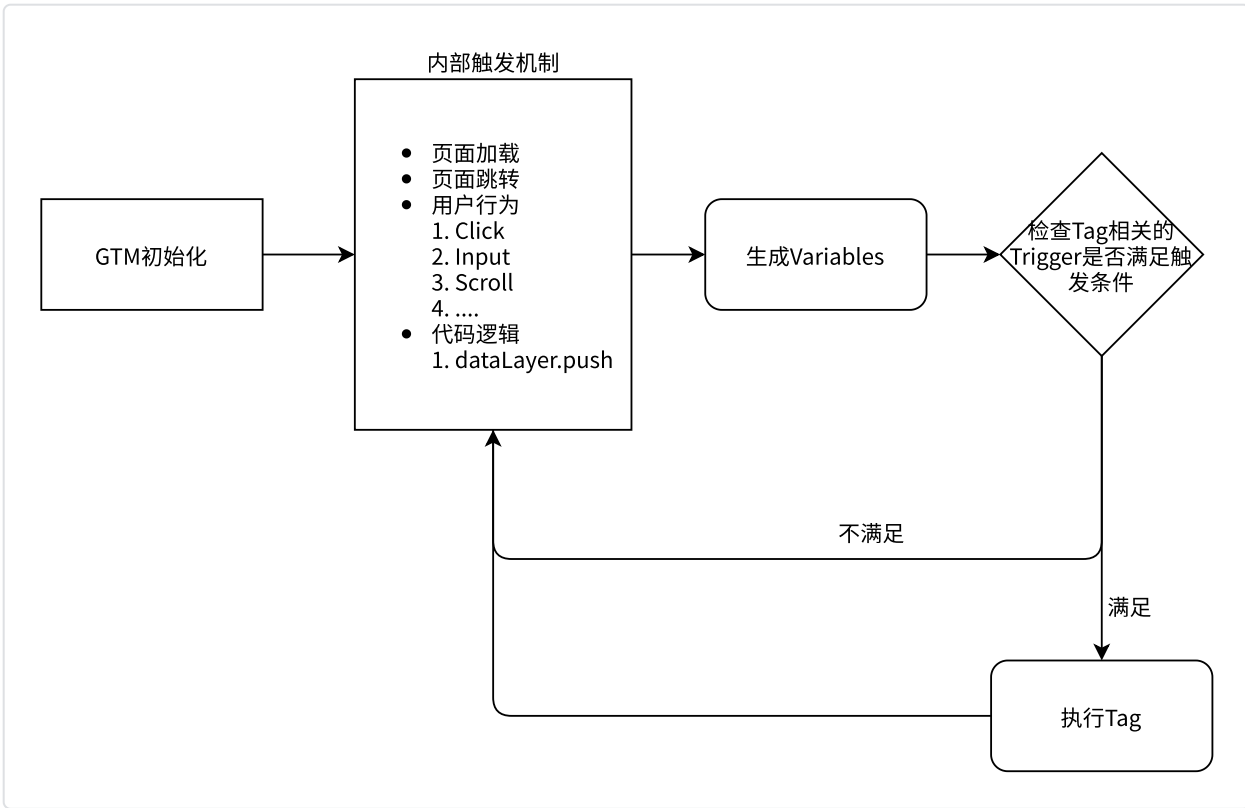
#### HTML

```
1 <script>
2   // 这里有一个放window上的javascript变量, 用在下图的GTM配置里面
3   window.userId = Math.round(Math.random(0, 10) * 10) % 10
4 </script>
```

Variable Type

Global Variable Name

## 对GTM生命周期的理解



## 对Variables的理解

一般情况下, Variables都在Trigger触发前生成, 并且不会因为Tag的执行完毕, 去销毁这些Variables

### Trigger Configuration

Trigger Type: **Timer**

Event Name: **gtm.timer**

Interval: **2000**

Limit: **1**

Enable this trigger when all of these conditions are true

**userId** greater than 0

### Tag Configuration

Tag Type: **custom-event-template** (Custom Template)

Tag permissions: **Tag permissions**

参数1: **{{all params}}**

### Triggering

Firing Triggers: **timer** (Timer)

### Variable Configuration

Variable Type: **Custom JavaScript**

Custom JavaScript

```

1 function(){
2   return {
3     'All Measure Ids': {{All Measure Ids}},
4     'custom-event-arg1': {{custom-event-arg1}},
5     'form element': {{form element}},
6     'userId': {{userId}},
7     'Click Element': {{Click Element}},
8     'Click ID': {{Click ID}}
9   }
10 }

```

### Sandboxed JavaScript

Before you start...

Sandboxed JavaScript is a limited subset of JavaScript that allows Tag Manager's custom templates to operate safely and securely. [Learn more](#)

```

1 // Enter your template code here.
2 const log = require('logToConsole');
3 log('data = ', data);
4
5 // Call data.gtmOnSuccess when the tag is finished.
6 data.gtmOnSuccess();

```

### Template Preview

#### Tag Configuration

Tag Type: **custom-event-template** (Custom Template)

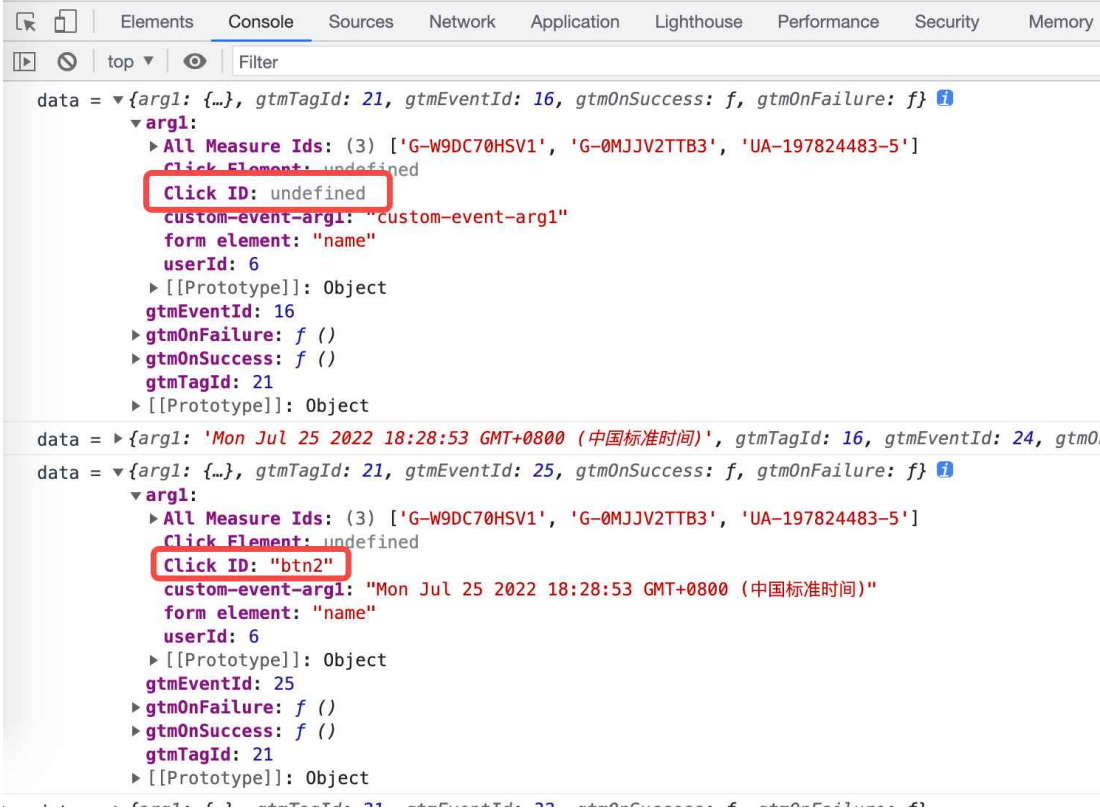
Tag permissions: **Tag permissions**

参数1: **{{custom-event-arg1}}**

click(trigger all)

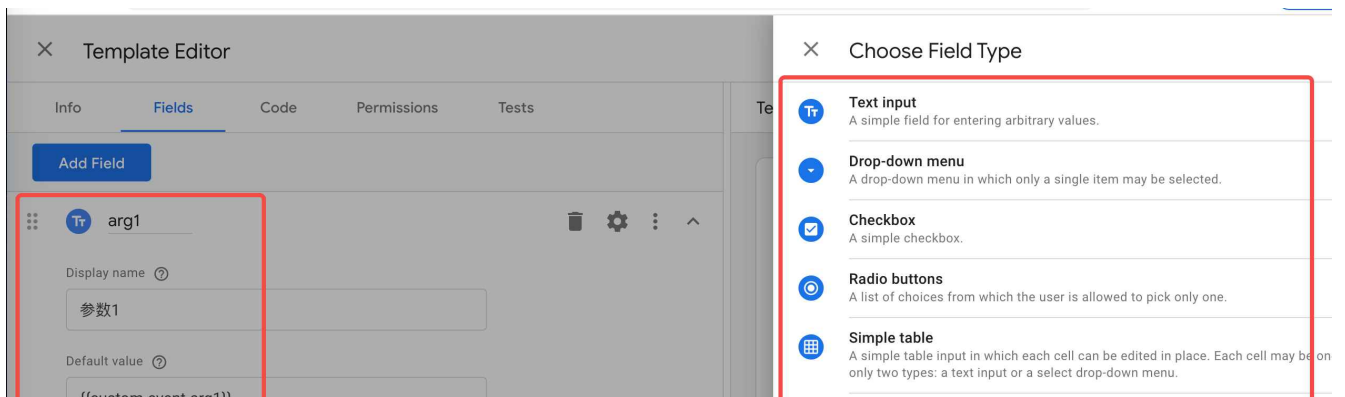
click(trigger test1)

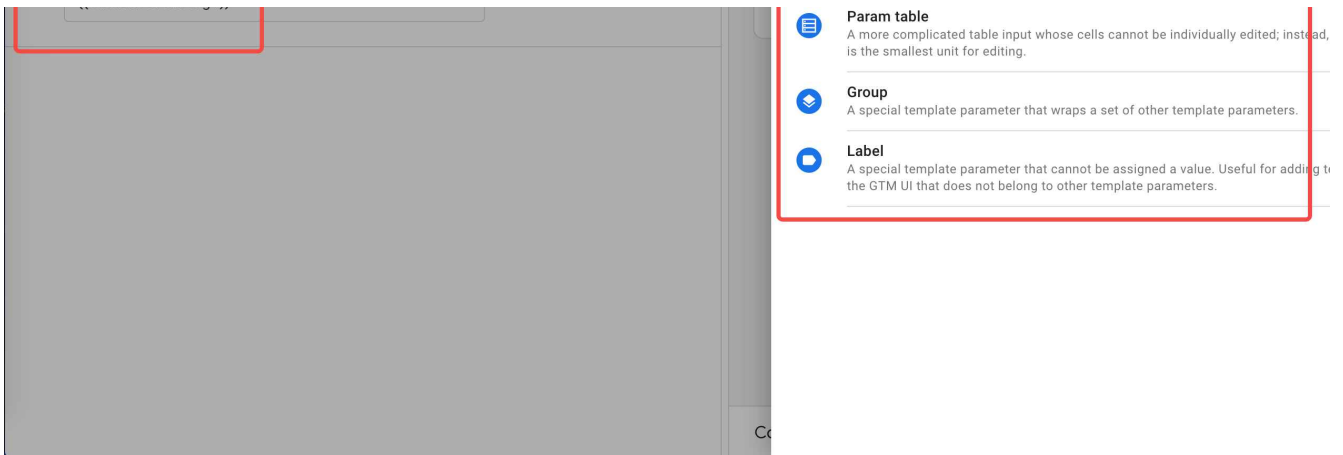
```
trigger btn1Click();
```



## 代码模板 (Tag Template)

- Template和Tag的关系, 有点像类和实例
  - 一个类通过构造函数传入参数可以得到实例
  - 同样: Tag通过指定参数和Template模板就能做具体的任务
- Template参数定义
  - 左侧是参数列表
  - 右侧是参数类型





- Template代码实现

```
Sandboxed JavaScript Documentation  
  
i Before you start... ×  
Sandboxed JavaScript is a limited subset of JavaScript that allows Tag Manager's custom templates to operate safely and securely. Learn more  
  
1 // Enter your template code here.  
2 const log = require('logToConsole');  
3 const copyWindow = require('copyFromWindow');  
4  
5 log('data =', data);  
6 var userid = copyWindow('userid');  
7  
8 // Call data.gtmOnSuccess when the tag is finished.  
9 data.gtmOnSuccess();
```

- 这个代码是沙箱环境执行的
- 一些脚本方法需要先引用API, 然后还要配置相关的权限, 之后才能正常执行
- 这个环境里面无法拿到Date对象, 但是可以通过getTimestampMillis拿到时间戳