

Get API Key

To use the Google Maps JavaScript API, you must register your app project on the Google API Console and get a Google API key which you can add to your app.

Quick guide to getting a key

Step 1: Get an API Key from the Google API Console

Click the button below, which guides you through the process of registering a project in the Google API Console, activates the Google Maps JavaScript API and any related services automatically, and generates a generic, unrestricted API key.

GET A KEY

Notes:

- **Tip:** During development and testing, you can register a project for testing purposes in the Google API Console and use a generic, unrestricted API key. When you are ready to move your app or website into production, register a separate project for production, create a browser-restricted API key, and add the key to your application.
- **Premium Plan customers:** For production-ready apps, you must use a browser-restricted API key that is set up in the Google Maps APIs Premium Plan project created for you when you purchased the Premium Plan. Alternatively, you can use a client ID (#client-id) in combination with URL registration (instead of an API key).
- For more information, see the detailed guides below for Standard Plan (#standard-auth) and Premium Plan (#premium-auth) customers.

Step 2: Add the API key to your application

When loading the Google Maps JavaScript API, substitute YOUR_API_KEY in the code below with the API key you got from the previous step.

```
<script async defer src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap" type="text/javascript"></script>
```

More about API keys

- The API key allows you to monitor your application's API usage in the Google API Console. See Google API Console help (https://support.google.com/googleapi/#topic=7013279) for more information.
- If you are a Standard Plan customer, with an API key you have access to generous free daily quota (https://developers.google.com/maps/documentation/javascript/usage), as well as the option to increase your daily quota by enabling pay-as-you-go billing (https://developers.google.com/maps/documentation/javascript/usage#increase_quota).
- If you are a Premium Plan customer, you must use an API key or your client ID to access all the custom features and benefits of your Premium Plan.
- Registering for an API key ensures that Google can contact you about your application if necessary.

Detailed guide for users of the standard Google Maps JavaScript API

Follow these steps to get an API key:

1. Go to the Google API Console (https://console.developers.google.com/flows/enableapi?apiid=maps_backend,geocoding_backend,directions_backend,distance_matrix_backend,elevation_backend,places_backend&reusekey=true).
2. Create or select a project.
3. Click **Continue** to enable the API and any related services.

4. On the **Credentials** page, get an **API key**.
Note: If you have an existing unrestricted API key, or a key with browser restrictions, you may use that key.
5. From the dialog displaying the API key, select **Restrict key** to set a browser restriction on the API key.
6. In the **Key restriction** section, select **HTTP referrers (web sites)**, then follow the on-screen instructions to set referrers.
7. (Optional) Enable billing. See [Usage Limits](https://developers.google.com/maps/documentation/javascript/usage) (https://developers.google.com/maps/documentation/javascript/usage) for more information.

In the Google API Console, you can also [look up an existing key](https://console.developers.google.com/project/_/apiui/credential) (https://console.developers.google.com/project/_/apiui/credential) or [view a list of enabled APIs](https://console.developers.google.com/project/_/apiui/enabled) (https://console.developers.google.com/project/_/apiui/enabled).

For more information on using the Google API Console, see [API Console Help](https://support.google.com/googleapi) (https://support.google.com/googleapi).

Detailed guide for users of the Google Maps APIs Premium Plan license

When using the Google Maps JavaScript API with a [Google Maps APIs Premium Plan license](https://developers.google.com/maps/premium/overview) (https://developers.google.com/maps/premium/overview), you must authenticate your application with either an API key or a client ID.
Note: If you have a [previous Maps API for Business](#) (#prev-license) license, you must use a client ID, not an API key.

Choosing an authentication method for your application

The section below provides a summary of the various tools and reports that are available to Premium Plan customers, based on the method you choose to authenticate your application.

- **Authentication using an API key**
(**Note:** Customers with a current Premium Plan license may use an API key, but customers holding a [previous Maps API for Business](#) (#prev-license) license must use a client ID.)
By using an API key to authenticate your applications, you can:
 - Manage all of your APIs in the [Google API Console](https://console.developers.google.com/project/_/apis/enabled) (https://console.developers.google.com/project/_/apis/enabled).
 - Access real-time usage data and 30 days of historical usage data for your application in the [Google API Console](https://console.developers.google.com/project/_/apis/enabled) (https://console.developers.google.com/project/_/apis/enabled).
 - View usage reports with more than 30 days of data in the [Google Cloud Support Portal](https://google.secure.force.com/) (https://google.secure.force.com/).
- **Authentication using a Client ID**
By using your client ID (instead of an API key) to authenticate your application, you can:
 - Add the `channel` parameter to requests so you can view more detailed usage reports.
 - View usage reports with more than 30 days of data in the [Google Cloud Support Portal](https://google.secure.force.com/) (https://google.secure.force.com/).
 - Use Maps Analytics tools for the Maps JavaScript API.

Get more information on [reports available to Premium Plan customers](https://developers.google.com/maps/premium/reports/) (https://developers.google.com/maps/premium/reports/).

Note: The information below on using an API key applies only to the Google Maps APIs Premium Plan, which became available on **January 6, 2016**.

Have a *previous* Maps APIs for Work or Maps API for Business license? See our [Maps APIs for Work Licenses guide](https://developers.google.com/maps/premium/previous-licenses/) (https://developers.google.com/maps/premium/previous-licenses/). To determine whether you have a previous license: In the [Google Cloud Support Portal](https://google.secure.force.com/) (https://google.secure.force.com/), click **Maps: Usage Report** on the left. If the ID at the top of the report is in the following format, you have the new Premium Plan:
`gme-[company] & proj-[number] ([type])`
Otherwise, you have a previous license.

Authenticating your application using an API key

Follow these steps to get an API key:

1. Go to the [Google API Console](https://console.developers.google.com/flows/enableapi?apiid=maps_backend,geocoding_backend,directions_backend,distance_matrix_backend,elevation_backend,places_backend&reusekey=true)
(https://console.developers.google.com/flows/enableapi?apiid=maps_backend,geocoding_backend,directions_backend,distance_matrix_backend,elevation_backend,places_backend&reusekey=true)
.
2. From the Project drop-down menu, select the project created for you when you purchased the Premium Plan. The project name starts with **Google Maps APIs for Business** or **Google Maps for Work** or **Google Maps**. **Important:** If you have a [previous Maps API for Business](#) (#prev-license) license, you must use a client ID, not an API key.
3. Click **Continue**.
4. On the **Credentials** page, get an **API key**.
Note: If you have an existing unrestricted API key, or a key with browser restrictions, you may use that key.
5. From the dialog displaying the API key, select **Restrict key** to set a browser restriction on the API key.
6. In the **Key restriction** section, select **HTTP referrers (web sites)**, then follow the on-screen instructions to set referrers.

You can also [look up an existing key](https://console.developers.google.com/project/_/apiui/credential) (https://console.developers.google.com/project/_/apiui/credential) in the Google API Console.

For more information on using the Google API Console, see [API Console Help](https://support.google.com/googleapi) (<https://support.google.com/googleapi>).

Authenticating your application using a client ID

Upon purchasing your [Google Maps APIs Premium Plan license](https://developers.google.com/maps/premium/overview) (<https://developers.google.com/maps/premium/overview>), you will receive a welcome email from Google that contains your **client ID**, which allows to access the [various reports and tools of the Google Maps APIs Premium Plan](#) (#clientID-features).

All client IDs begin with a **gme-** prefix. Below is the format of a typical client ID.

gme-[company] & proj-[number] ([type])

You can authenticate your application with the Google Maps JavaScript API using a client ID in combination with **URL registration** (instead of an API key).

Note: This client ID is not a key, and it can only be used by URLs that you authorize. Read more about [registering authorized URLs](#) (#registering-authorized-urls).

Specifying a client ID when loading the API

The code below shows you how to substitute YOUR_CLIENT_ID with your own client ID when loading the Google Maps JavaScript API.

```
<script async defer
src="https://maps.googleapis.com/maps/api/js?client=YOUR_CLIENT_ID&v=3.28&callback=initMap"></script>
```

You must specify the [release version](https://developers.google.com/maps/documentation/javascript/versions) (<https://developers.google.com/maps/documentation/javascript/versions>) (also referred to as the **feature-stable** version) or an earlier version, by appending a `v=3.28` parameter. Applications that use the experimental version are not covered under the Google Maps APIs Premium Plan SLA.

Note: If you were previously using an API key for authentication and are switching to using a client ID, you must remove the **key** parameter before loading the API. **The API will fail to load if both a client ID and an API key are included.**

Restricting an API key

Google Maps APIs are available for web browsers, Android or iOS apps, and via HTTP web services. APIs in any platform can use a generic (unrestricted) API key. You can optionally add a restriction (for example, HTTP referrer) to the API key. Once restricted, a key will only work on platforms that support that type of restriction.

Tip: Before moving your app or website to production, you should secure your API key. Keys for the Google Maps JavaScript API use the **HTTP referrers (web sites)** key restriction. [Learn more about keys and credentials](#) (<https://developers.google.com/maps/faq#keysystem>).

To add web browser restrictions to an existing, generic API key, do the following:

1. Go to the Credentials page of the [Google API Console](https://console.developers.google.com/project/_/apiui/credential) (https://console.developers.google.com/project/_/apiui/credential).
2. Select the project that contains the API key you want to edit.
3. On the **Credentials** page, from the list of API keys, select the name of the API key to edit the details of the key.
4. In the **Key restriction** section of the page, select **HTTP referrers (web sites)**, follow the on-screen instructions to set referrers, then click **Save**.

Note: file:// referers need a special representation to be added to the Key restriction. The "file:/" part should be replaced with "__file_url__" before being added to the Key restriction. For example, "file:///path/to/" should be formatted as "__file_url__//path/to/*". After enabling file:// referers, it is recommended you regularly check your usage, to make sure it matches your expectations.

Registering authorized URLs

To prevent a third party from using your client ID on their own website, the use of your client ID is restricted to a list of URLs that you specifically authorize.

To see the URLs you have already authorized or to authorize additional URLs:

1. Log in to the [Google Cloud Support Portal](https://google.secure.force.com/) (https://google.secure.force.com/).
2. In the left-hand menu, click **Maps: Manage Client ID**.

You can add up to 100 URLs at a time, to a total of 3000 URLs. If you require higher limits, please contact [support](https://developers.google.com/maps/premium/support) (https://developers.google.com/maps/premium/support).

The following considerations apply regarding URLs that are authorized:

The domain name or IP address does not have to be publicly accessible.

For example, `http://myintranet` and `http://192.168.1.1` are valid entries.

All subdomains of a specified domain are also authorized.

For example, if `http://example.com` is authorized, then `http://www.example.com` is also authorized. The reverse is not true: if `http://www.example.com` is authorized, `http://example.com` is not necessarily authorized.

All subpaths of an authorized path are also authorized.

For example, if `http://example.com` is authorized, then `http://example.com/foo` is also authorized. In addition, because subdomains of a specified domain are also authorized, `http://sub.example.com/bar` is authorized.

Paths are case sensitive.

For example, `http://www.example.com/ThisPath/` is not the same as `http://www.example.com/thispath/`.

You may restrict valid URLs to those using certain ports.

For example, if `http://example.com:8080/foo` is specified, that doesn't authorize `http://example.com`.

HTTP and HTTPS protocols are considered different URLs.

For example, if `https://example.com` is authorized, `http://example.com` is not necessarily authorized. If you'd like to authorize both at once, you may add a domain without using a protocol: `example.com/`

All the rules above are applied to each authorization, so you should take care to plan your authorizations carefully. For example, because all subpaths of a specified path are authorized, and all subdomains, you may end up authorizing pages that you didn't intend to. For example:

`http://example.com/` also authorizes `http://sub.example.com/path`.

For more information, see [Troubleshooting Google Maps APIs Premium Plan Authorization](https://developers.google.com/maps/premium/troubleshooting-auth) (https://developers.google.com/maps/premium/troubleshooting-auth/).

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](http://creativecommons.org/licenses/by/3.0/) (http://creativecommons.org/licenses/by/3.0/), and code samples are licensed under the [Apache 2.0 License](#)

(<http://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated June 7, 2017.



Stack Overflow

Ask a question under the google-maps-api-3 tag.



GitHub

Fork our samples and try them yourself.



Blog

Read the latest updates, customer stories, and tips.



Issue Tracker

Something wrong? Send us a bug report!