

# Google Maps API

CodeIgniter Library

Version: 1.1

Author: BIOSTALL (Steve Marks)

Website: <http://biostall.com>

Email: [info@biostall.com](mailto:info@biostall.com)

## Introduction

This CodeIgniter library provides an easy way to display simple maps within an application/website using the Google Maps V2 API. It allows maps to be shown, including customisable markers, with just a few lines of code.

## Installation

In order to use the library you will need to download the Googlemaps.php file and place it in the 'libraries' directory of your application folder. Following the demonstrations and documentation below you will then be able to begin creating maps right away.

## The Basics

Once the installation instructions above are complete there are just two things we need to do in order to create a map. The first is to amend our controller in order to initialize and customise our output, and the second is to include two lines of code in our view where we want the map to be displayed:

### The Controller

The example below shows how to create a simple map with all the default controls and no markers:

```
// Load the library
$this->load->library('googlemaps');

// This is our Google API key. You will need to change this for your own website
$config['apikey'] = 'ABQIBAAADiypSsSGMm4zOSm7T0Nf1BT2yXp_ZAY8_ufC3CFXhHIE1NvwkxTegq9k46u7yla9Jc-SjozzP0J8ig';

// Initialize our map, passing in any configuration values we have set
$this->googlemaps->initialize($config);

// Create the map. This will return the Javascript to be included in our pages <head> section and the HTML code to be
// placed where we want the map to appear.
$data['map'] = $this->googlemaps->create_map();

// Load our view, passing the map data that has just been created
$this->load->view('my_view', $data);
```

### The View

There are two things we need to add to our view. The first is the Javascript which should be placed in the <head> section of your website as follows:

```
<head>
<?php echo $map['javascript']; ?>
</head>
```

The next thing is the actual map. This should be placed where you want the map to appear as follows:

```
<?php echo $map['mapdiv']; ?>
```

## Customizing the Map

The library allows you to adjust the way your map appears by passing in additional \$config values to \$this->googlemaps->initialize(\$config). There are as follows:

Name	Type	Default	Possible Values	Description
map_div_id	string	'map_canvas'		Sets the ID of the div in which the map is displayed.
map_width	string	'100%'		Sets the map width. If no unit is included it will default to 'px'
map_height	string	'450px'		Sets the map height. If no unit is included it will default to 'px'
center_lat_long	string	'37.4419, -122.1419'		The latitude and longitude of the center of the default map view
center_address	string			As an alternative to centering the map based on a latitude and longitude, you can use an address instead.
zoom	integer	13	0-22 depending on location.	The default zoom level where 0 is the coarsest
type	string	'normal'	'normal', 'satellite', 'hybrid', 'earth'	The type of aerial view to display
controls	array		'GLargeMapControl3D', 'GLargeMapControl', 'GSmallMapControl', 'GSmallZoomControl3D', 'GSmallZoomControl', 'GScaleControl', 'GMapTypeControl', 'GHierarchicalMapTypeControl', 'GOverviewMapControl', 'GNavLabelControl'	One or more controls to display on the map
hide_controls	boolean	FALSE	TRUE or FALSE	If no 'controls' are not set, turning this to FALSE will hide the default controls
icon_image	string			A URL to the image to be used for all marker icons if markers are placed on the map.
google_host	string	'maps.google.com'		If using the map to display markers in a certain country, you will get better results if you change this to the country in question. For example, if only displaying markers in Germany, change this value to 'maps.google.de'
sensor	boolean	FALSE	TRUE or FALSE	Turn to TRUE if the device on which the map is being displayed contains tracking functionality.

## Adding Markers

The library also allows you to add multiple markers to the map at specified positions. To add a single marker we can add the following code before the `create_map()` function is called:

```
// Set the marker parameters as an empty array. Especially important if we are using multiple markers
$marker = array();

// Specify the address for where the marker should appear. Alternatively we can use the lat_long value
$marker['address'] = 'Crescent Park, Palo Alto';

// Once all the marker parameters have been specified lets add the marker to our map
this->googlemaps->add_marker($marker);
```

To create multiple markers simply duplicate the above code the required amount of times.

Like the map itself, we can also specify a number of parameters for individual markers to change how and where they appear. These parameters are as follows:

Name	Type	Default	Possible Values	Description
lat_long	string			The latitude and longitude at which to show the marker
address	String			As an alternative to positioning markers based on a latitude and longitude, you can use an address instead.
infowindow_type	string		'text' or 'html'	If you want to show an infowindow/bubble you can specify whether to use a text or HTML version. The HTML version allows HTML tags, styling and images.
infowindow_content	string			If 'infowindow_type' is set, this is the text/HTML to show in the infowindow/bubble.
infowindow_open	Boolean	FALSE	TRUE or FALSE	Whether to open this markers infowindow/bubble by default when the page loads. If multiple markers have this set as TRUE, only the first infowindow will open by default.
icon_image	string			A URL to the image to be used for this marker. If 'icon_image' is set on the map, the marker 'icon_image' will take priority.

To leave feedback, ask questions or report bugs please contact me at [info@biostall.com](mailto:info@biostall.com) or leave a comment at <http://biostall.com>.