

Searching With Google Places

Nearly every business, small and large, has a listing in [Google Places](#). You would be most familiar with Google Places just by searching within Google Maps. Using Google Places requires some JavaScript code, but ViziApps makes this easy by providing a ready-to-use library.

This document describes how ViziApps gives you direct access to search and get results from Google Places. You can find out how to show the search results on a map in the companion document, [Integrating Google Maps](#).

Load the [template app](#) called **Google Places and Maps** to see the following examples in action.



Set up the library

The first step is to install the ViziApps JavaScript library into your app's HTML Header. Use the document event **GooglePlacesLoaded** to guarantee that calls are only made after the library is installed and ready.

```
<script
src="https://s3.amazonaws.com/viziapps/apps/google_places_1.0.2.js"></script>

<script>
var goomaps;
$(document).on( 'GooglePlacesLoaded', function()
{
    // create an instance of the google maps api:
    goomaps = new v.googlePlaces();
});

function findBakeries() {
    // make calls to the google maps library, such as:
    if ( !goomaps ) return;
    goomaps.find( 'bakeries' )
    // ...
}
</script>
```

Search Google Places

Search Google Places by calling the **.find(*searchTerm*, *optionSet*)** method. Process the success or failure results with the **jQuery Deferred Methods** **.done()** and **.fail()**.

- *searchTerm* is a string, such as the name of a store.
- *optionSet* is an optional object that may contain several options, described below.

```
var searchTerm = v.getFieldValue( 'searchTextField' );
var gpsloc = {
    latitude: v.getFieldValue( 'gpsLatitude' ),
    longitude: v.getFieldValue( 'gpsLongitude' )
};
var optionSet = {
    location: gpsloc,
    pageLimit: 1
};

goomaps.find( searchTerm, optionSet )
.fail( function( errorMessage )
{
    alert( 'Search error: ' + errorMessage );
} )
.done( function( placeList )
{
    console.log( JSON.stringify( placeList, null, '  ' ) );
} );
```

Place Objects

The return *placeList* in the **.done()** method is an array of Place Objects, each containing descriptive information such as the place's name, address, coordinates, etc. The list of properties is fully documented: **Place Search Results**.

Using Location

- If the *optionSet* includes a location, then the search will only return places that are nearby that location.
- If there is no location specified, then any search context will be based on the *searchTerm*, for example “pizza in New York”.

Options for Searching

The following properties may be included in the *optionSet* parameter for **.find()**. If the **types** property is included, then the *searchTerm* parameter may be set to null or empty.

location	string or object	Location specified as a lat,lng string, e.g. '42.360070,-71.061105', or as an object with lat/lng or latitude/longitude properties.
radius	integer	Number of kilometers near the location to prioritize the search results. Only used if location is provided.
types	string list or array	Example: 'lodging,shopping'. For the full list, see: Supported Types . You can also get an array with the types list from the library: goomap.placetypes .
validate	true or false	If true, rejects any search results that do not contain the words in the <i>searchTerm</i> .
filter	string list or array	The search results are filtered to guarantee that each place contains at least one of the filter terms.
minprice	integer	0 to 4, only returns places of this price level or greater.
maxprice	integer	0 to 4, only returns places of this price level or lower.
minrating	number	0.0 to 5.0, only returns places better than this average rating.
opennow	true or false	Only returns places that are open right now.
pagelimit	integer	1 to 3; Google Places returns 20 search results per page, by default up to 3 pages with a minimum 2-second delay between pages. Reduce the processing delay and size of the search results by limiting the page count.

Request Specific Places

After making a search with **.find()**, use the **.select()** method to retrieve a specific list or subset of the search results. There are several ways to make this call:

.select()	Called without any parameters, the return value would be exactly the same array as returned in the original call to .find() .
.select(index)	Returns a single Place Object (not an array) at the specified index in the list.
.select(byProperty, matchValue)	Returns the first Place Object that has its property <i>byProperty</i> exactly matching <i>matchValue</i> .

Get Place Details

Each Place Object returned from **.find()** gives a summary set of information about each place, but more detailed information is available. Use the **.details(placeObject)** method to get these details about a specific place.

```
goomap.find( 'bakeries in Boston' )
.done( function( placeList )
{
    var details = goomap.details( placeList[0] );
    console.log( JSON.stringify( details, null, '  ' ) );
    v.setFieldValue( 'placeImageField', details.mainphoto );
});
```

The detailed properties that are available are fully documented: [Place Details Responses](#). The ViziApps library adds one property: if at least one photo is available, a **mainphoto** property will contain a URL to the first photo in the list.

From:

<https://viziapps.com/dokuwiki/> - **ViziApps Help Wiki**

Permanent link:

https://viziapps.com/dokuwiki/searching_with_google_places

Last update: **2015/01/14 12:04**

