GRAB A BITE!

**Google places API: to find “open” “restaurants” “near me”**

**Key**- AIzaSyCvkPVO9PN9rc1Q8-frKtGD\_rMrn2fKVAk

To find restaurants near me:

var request = {

location: pyrmont,

types: ['restaurant']

};

infowindow = new google.maps.InfoWindow();

places = new google.maps.places.PlacesService(map);

places.nearbySearch(request, callback);

**Parameters** in Google places API

key — Your application's API key. This key identifies your application. See Get a key for more information. (**Key**- AIzaSyCvkPVO9PN9rc1Q8-frKtGD\_rMrn2fKVAk)

input — The text input specifying which place to search for (for example, a name, address, or phone number).

inputtype — The type of input. This can be one of either textquery or phonenumber.

**Optional parameters:**

**Contact**

The Contact category includes the following field: opening\_hours (Place Search returns only open\_now; use a Place Details request to get the full opening\_hours results).

Find Place examples

The following example shows a Find Place request for "Museum of Contemporary Art Australia", including the photos,formatted\_address, name, rating, opening\_hours, and geometry fields:

https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=Museum%20of%20Contemporary%20Art%20Australia&inputtype=textquery&fields=photos,formatted\_address,name,rating,opening\_hours,geometry&key=YOUR\_API\_KEY

The following example shows a Find Place request for "Mongolian Grill", using the locationbias parameter to prefer results within 2000 meters of the specified coordinates:

https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=mongolian%20grill&inputtype=textquery&fields=photos,formatted\_address,name,opening\_hours,rating&locationbias=circle:2000@47.6918452,-122.2226413&key=YOUR\_API\_KEY

## **Nearby Search requests**

Earlier versions of the Places API referred to Nearby Search as Place Search.

A Nearby Search lets you search for places within a specified area. You can refine your search request by supplying keywords or specifying the type of place you are searching for.

A Nearby Search request is an HTTP URL of the following form:

https://maps.googleapis.com/maps/api/place/nearbysearch/*output*?*parameters*

where output may be either of the following values:

* json (recommended) indicates output in JavaScript Object Notation (JSON)
* xml indicates output as XML

Certain parameters are required to initiate a Nearby Search request. As is standard in URLs, all parameters are separated using the ampersand (&) character.

**Required parameters**

* key — Your application's [API key](https://support.google.com/googleapi/answer/6158862). This key identifies your application. See [Get a key](https://developers.google.com/places/web-service/get-api-key) for more information.
* location — The latitude/longitude around which to retrieve place information. This must be specified aslatitude,longitude.
* radius — Defines the distance (in meters) within which to return place results. The maximum allowed radius is 50 000 meters. Note that radius must not be included if rankby=distance (described under **Optional parameters**below) is specified.
* If rankby=distance (described under **Optional parameters** below) is specified, then one or more of keyword, name, or type is required.

**Optional parameters**

* opennow — Returns only those places that are open for business at the time the query is sent. Places that do not specify opening hours in the Google Places database will not be returned if you include this parameter in your query.

#### **Text search examples**

**Note:** In these examples, you need to replace the key with your own API key in order for the request to work in your application.

**Example 1:** The following example shows a search for restaurants near Sydney.

https://maps.googleapis.com/maps/api/place/textsearch/xml?query=restaurants+in+Sydney&key=***YOUR\_API\_KEY***

**Example 2:** The following example shows a search for an incomplete address, in this case, a street address that does not include a city or state or country.

https://maps.googleapis.com/maps/api/place/textsearch/json?query=123+main+street&key=***YOUR\_API\_KEY***

**Example 3:** The following example shows a search for the same incomplete address in sample 2, and includes location and radius parameters to bias the results to a region of interest. Compare the results of sample 2 to sample 3.

https://maps.googleapis.com/maps/api/place/textsearch/json?query=123+main+street&location=42.3675294,-7

**Latitude and longitude of current location:**

GeoCoder

**Google Directions API(to calc time to the restaurants returned):**

**Note: traffic not considered as Google direction API, distance matrix api etc are paid.**

public double GetDistanceBetweenPoints(double lat1, double long1, double lat2, double long2)

{

double distance = 0;

double dLat = (lat2 - lat1) / 180\* Math.PI;

double dLong = (long2 - long1) / 180 \* Math.PI;

double a = Math.Sin(dLat / 2) \* Math.Sin(dLat / 2)

+ Math.Cos(lat1 / 180\* Math.PI) \* Math.Cos(lat2 / 180\* Math.PI)

\* Math.Sin(dLong/2) \* Math.Sin(dLong/2);

double c = 2 \* Math.Atan2(Math.Sqrt(a), Math.Sqrt(1 - a));

//Calculate radius of earth

// For this you can assume any of the two points.

double radiusE = 6378135; // Equatorial radius, in metres

double radiusP = 6356750; // Polar Radius

//Numerator part of function

double nr = Math.Pow(radiusE \* radiusP \* Math.Cos(lat1 / 180 \* Math.PI), 2);

//Denominator part of the function

double dr = Math.Pow(radiusE \* Math.Cos(lat1 / 180 \* Math.PI), 2)

+ Math.Pow(radiusP \* Math.Sin(lat1 / 180 \* Math.PI), 2);

double radius = Math.Sqrt(nr / dr);

//Calculate distance in meters.

distance = radius \* c;

return distance; // distance in meters

}

Final Example for us:

https://maps.googleapis.com/maps/api/place/nearbysearch/xml?query=restaurant+near+me&key=AIzaSyCvkPVO9PN9rc1Q8-frKtGD\_rMrn2fKVAk&location=17.440081,%2078.348915&rankby=distance&opennow

This will give us nearby places, given our latitude and longitude. Filter by <type> food </type>

Results-> geometry gives latitude and longitude of restaurant

Json: “types” is food