PLACE AUTO COMPLETE

This tutorial describes the usage of the Google Map API.

**EXAMPLE : Place Autocomplete using Google Map Api ( First Enable Place API )**

public class **AutocompleteSupportFragment** extends **Fragment**

Implements a fragment that presents a search box button to the user, which presents a search box UI when clicked. As the user types, the autocomplete service returns [Place](https://developers.google.com/places/android-sdk/reference/com/google/android/libraries/places/api/model/Place.html) predictions for places such as businesses, addresses and points of interest. When the user selects a place, the autocomplete service returns the response via the [PlaceSelectionListener](https://developers.google.com/places/android-sdk/reference/com/google/android/libraries/places/widget/listener/PlaceSelectionListener.html).

Google’s AutoComplete service is provided by Places SDK. AutoComplete api is used to auto populate places in response to the user search input. AutoComplete api suggests addresses, location, businesses as the user types and search.

Auto Complete can be added by adding auto complete widget or programatically to create your own user experience.

***Places autocomplete error fix 2019***

Hello folks how are you all,so today I will discuss an important topic about the recent upgradation of places SDK for android for which android programmer getting places autocomplete error.

So few days back while developing an app we faced an error Status, which is shown below



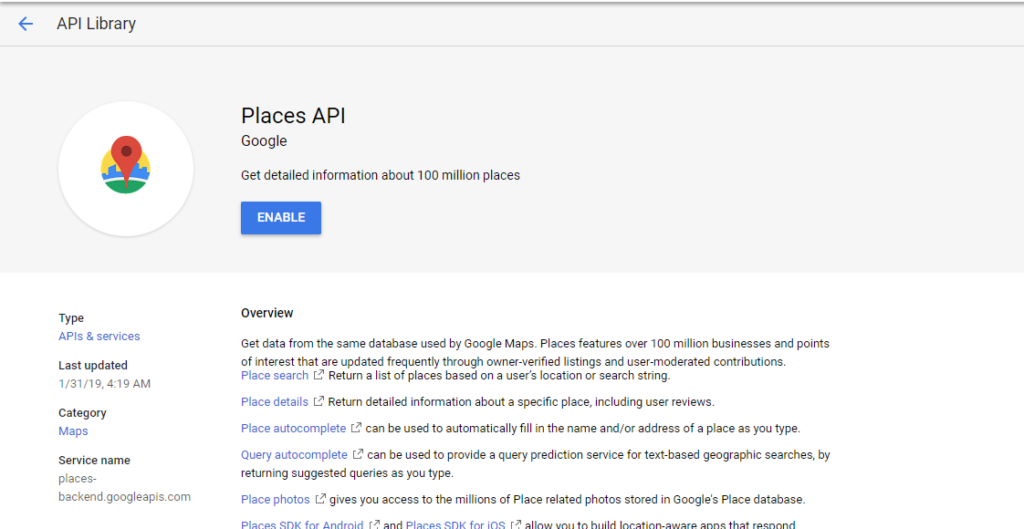
PLACES\_API\_ACCESS\_NOT\_CONFIGURED

As per [google](https://developers.google.com/places/android-sdk/client-migration) the current version of the places sdk for android(in Google Play services 16.0.0 ) is now replaced and will be completely turned off from July 29 onward.

Here we share with you the simple four step solution for the  ****places autocomplete****

**STEP 1 :**

So first you have to enable the Places API at google console(before there was a places SDK for android available)



**STEP 2 :**

Now we have to update our dependencies from

dependencies {

implementation 'com.google.android.gms:play-services-places:16.0.0'

}

To

dependencies {

implementation 'com.google.android.libraries.places:places:1.0.0'

}

**STEP 3 :**

Now we need to change some xml code at activity\_maps.xml from

<fragment  
 android:id="@+id/place\_autocomplete\_fragment"  
 android:name="com.google.android.gms.location.places.ui.PlaceAutocompleteFragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="10dp"  
 android:layout\_weight="5"/>

To

<fragment  
 android:id="@+id/autocomplete\_fragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:name="com.google.android.libraries.places.widget.AutocompleteSupportFragment"  
 android:layout\_margin="10dp"  
 android:layout\_weight="5"/>

**STEP 4 :**

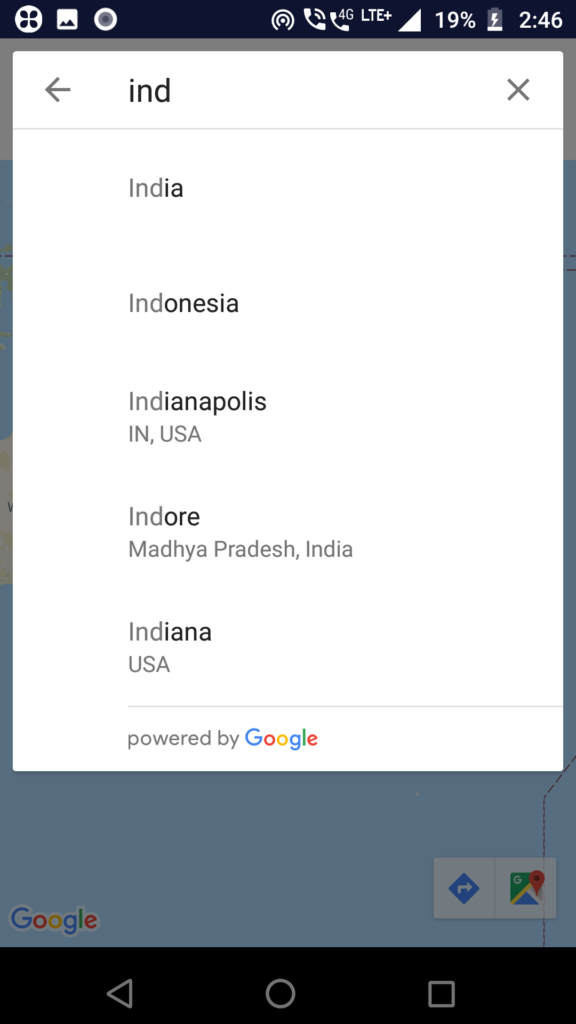
So now we have to update our MapsActivity.java from

*//Places Api*places = (PlaceAutocompleteFragment)getFragmentManager().findFragmentById(R.id.place\_autocomplete\_fragment);  
places.setOnPlaceSelectedListener(new PlaceSelectionListener() {  
 @Override  
 public void onPlaceSelected(Place place) {  
 mMap.clear();  
 mMap.addMarker(new MarkerOptions().position(place.getLatLng()).title(place.getName().toString()));  
 mMap.moveCamera(CameraUpdateFactory.*newLatLng*(place.getLatLng()));  
 mMap.animateCamera(CameraUpdateFactory.*newLatLngZoom*(place.getLatLng(), 12.0f));  
 destination = place.getAddress().toString();  
 destination = destination.replace("","+");  
 }  
  
 @Override  
 public void onError(Status status) {  
 Toast.makeText(MapsActivity.this, ""+status.toString(), Toast.*LENGTH\_SHORT*).show();  
 }  
});

To

if (!Places.*isInitialized*()) {  
 Places.*initialize*(getApplicationContext(), "your api key");  
 }  
  
*// Initialize the AutocompleteSupportFragment.* AutocompleteSupportFragment autocompleteFragment = (AutocompleteSupportFragment)  
 getSupportFragmentManager().findFragmentById(R.id.*autocomplete\_fragment*);  
  
 autocompleteFragment.setPlaceFields(Arrays.asList(Place.Field.*ID*, Place.Field.*NAME*));  
  
 autocompleteFragment.setOnPlaceSelectedListener(new PlaceSelectionListener() {  
 @Override  
 public void onPlaceSelected(Place place) {  
 *// TODO: Get info about the selected place.* Log.*i*(TAG, "Place: " + place.getName() + ", " + place.getId());  
 }  
  
 @Override  
 public void onError(Status status) {  
 *// TODO: Handle the error.* Log.*i*(TAG, "An error occurred: " + status);  
 }  
 });

now as you seen below the above error has gone now, and your app will now work as smooth as ever.



**STEP 1 :-** build.gradle ( Module: app ) file.

apply plugin: 'com.android.application'  
  
android {  
 compileSdkVersion 28  
 defaultConfig {  
 applicationId "com.google.auto.complete.places"  
 minSdkVersion 18  
 targetSdkVersion 28  
 versionCode 1  
 versionName "1.0"  
 testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"  
 resValue "string", "google\_maps\_key", (project.findProperty("GOOGLE\_MAPS\_API\_KEY") ?: "")  
 }  
 buildTypes {  
 release {  
 minifyEnabled false  
 proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
 }  
 }  
}  
  
dependencies {  
 implementation fileTree(dir: 'libs', include: ['\*.jar'])  
 implementation 'com.android.support:appcompat-v7:28.0.0'  
 implementation 'com.android.support.constraint:constraint-layout:1.1.3'  
 testImplementation 'junit:junit:4.12'  
 androidTestImplementation 'com.android.support.test:runner:1.0.2'  
 androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'  
  
 implementation 'com.android.support:design:28.0.0'  
 implementation 'com.google.android.libraries.places:places:1.0.0'  
}

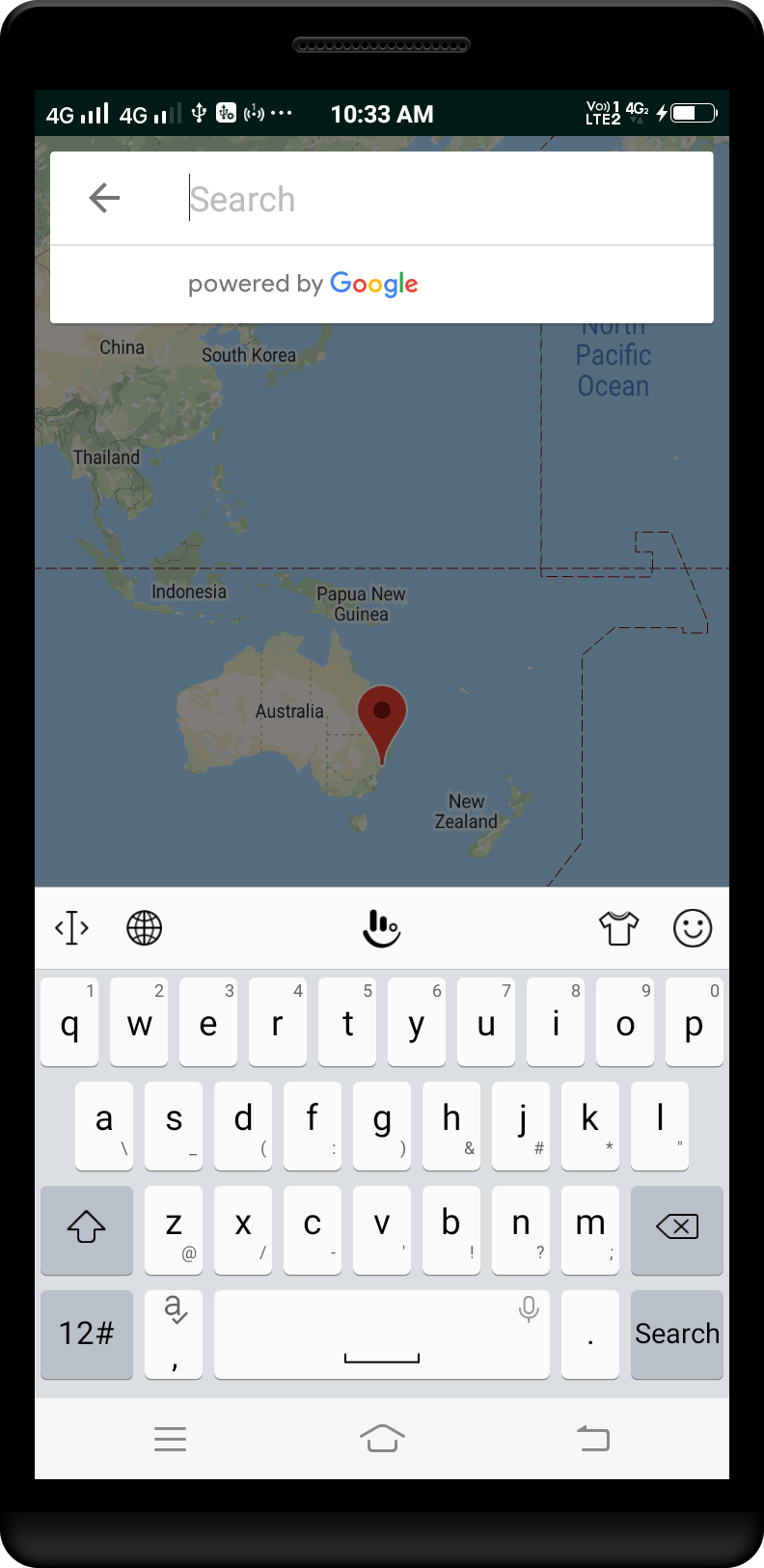
**STEP 2 :-** activity\_main.xml

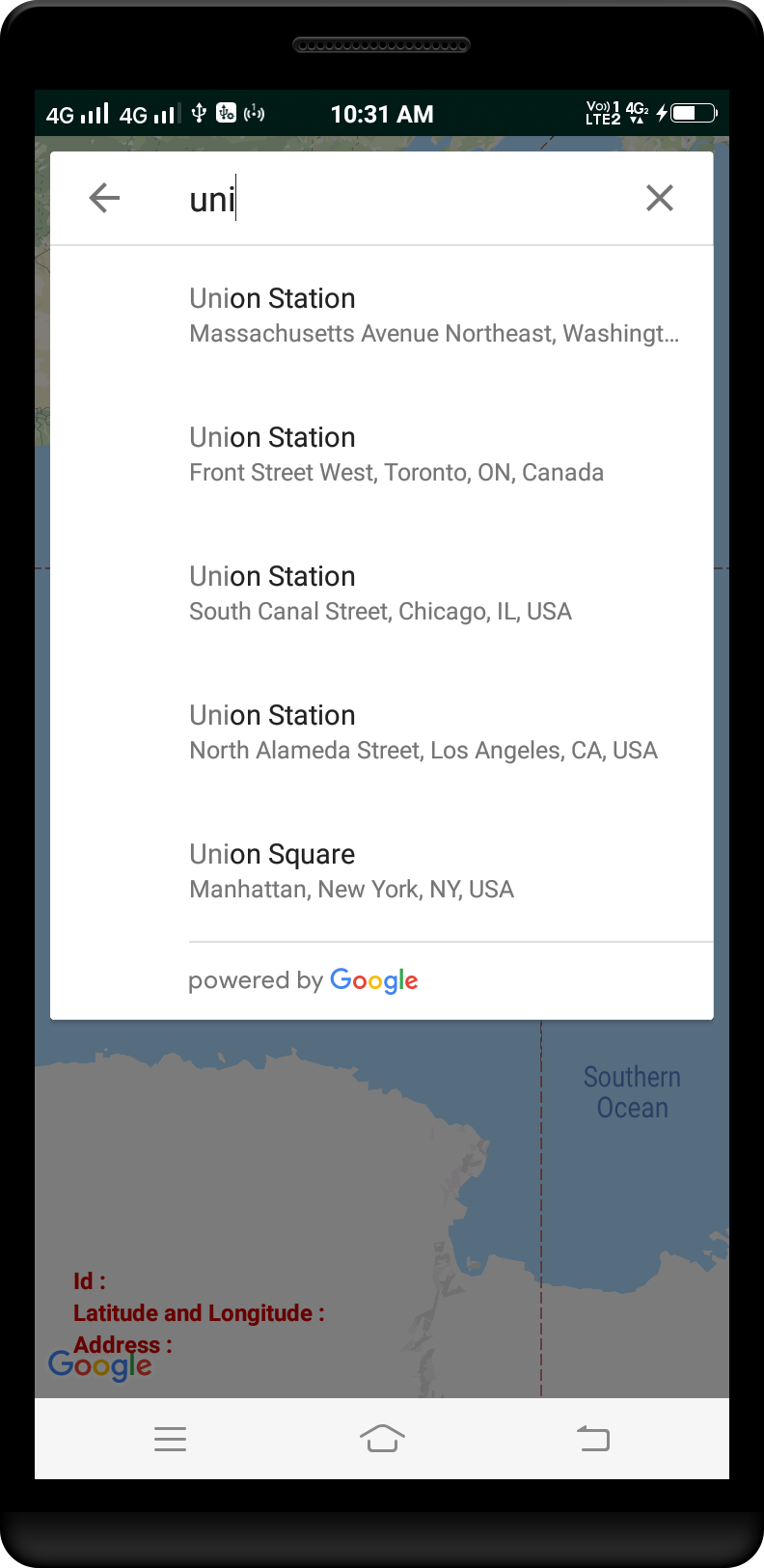
*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:orientation="vertical">  
  
 <FrameLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <fragment  
 android:id="@+id/map"  
 android:name="com.google.android.gms.maps.SupportMapFragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"/>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="10dp"  
 android:orientation="vertical">  
  
 <android.support.v7.widget.CardView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
  
 <fragment  
 android:id="@+id/autocomplete\_fragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:name="com.google.android.libraries.places.widget.AutocompleteSupportFragment"/>  
  
 </android.support.v7.widget.CardView>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_margin="20dp"  
 android:orientation="vertical"  
 android:gravity="left|bottom"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/id\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Id : "  
 android:textColor="@android:color/holo\_red\_dark"  
 android:textStyle="bold"  
 android:textSize="12dp"/>  
  
 <TextView  
 android:id="@+id/latitude\_Longitude\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Latitude and Longitude : "  
 android:textColor="@android:color/holo\_red\_dark"  
 android:textStyle="bold"  
 android:textSize="12dp"/>  
  
 <TextView  
 android:id="@+id/address\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Address : "  
 android:textColor="@android:color/holo\_red\_dark"  
 android:textStyle="bold"  
 android:textSize="12dp"/>  
 </LinearLayout>  
 </FrameLayout>  
</LinearLayout>

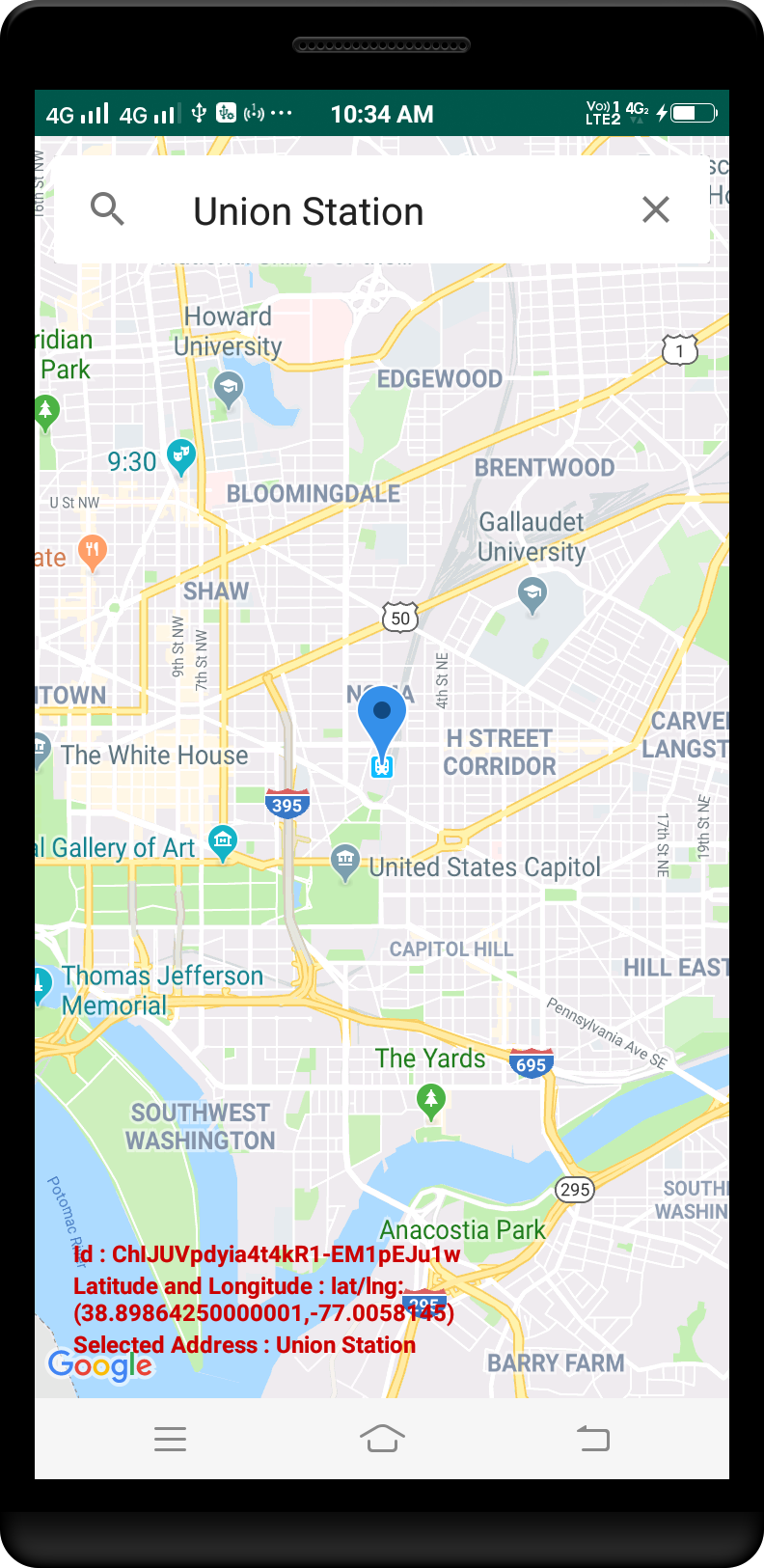
**STEP 5 :-** MainActivity.java

package com.google.auto.complete.places;  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.TextView;  
import com.google.android.gms.common.api.Status;  
import com.google.android.gms.maps.model.BitmapDescriptorFactory;  
import com.google.android.libraries.places.api.Places;  
import com.google.android.libraries.places.api.model.Place;  
import com.google.android.libraries.places.widget.AutocompleteSupportFragment;  
import com.google.android.libraries.places.widget.listener.PlaceSelectionListener;  
import java.util.ArrayList;  
import java.util.List;  
import android.support.v4.app.FragmentActivity;  
import android.widget.Toast;  
import com.google.android.gms.maps.CameraUpdateFactory;  
import com.google.android.gms.maps.GoogleMap;  
import com.google.android.gms.maps.OnMapReadyCallback;  
import com.google.android.gms.maps.SupportMapFragment;  
import com.google.android.gms.maps.model.LatLng;  
import com.google.android.gms.maps.model.MarkerOptions;  
  
public class MainActivity extends FragmentActivity implements OnMapReadyCallback {  
  
 private String TAG ="FRAGMENT\_AUTOCOMPLETE";  
 private static final String *GOOGLE\_PLACES\_API\_KEY* = "AIzaSyDVGtOq0Hq9f9oR9wDNVRxYQQzF6W1FYeI";  
 private GoogleMap googleMap;  
 private SupportMapFragment mapFragment;  
  
 */\*\*  
 \* UI Widgets.  
 \*/* private TextView idTextView;  
 private TextView latitudeAndLongitudeTextView;  
 private TextView addressTextView;  
 */\*\*  
 \* Labels.  
 \*/* private String id;  
 private String latitudeAndLongitudeLabel;  
 private String addressLabel;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 mapFragment = (SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.*map*);  
 assert mapFragment != null;  
 mapFragment.getMapAsync(this);  
  
 idTextView = findViewById(R.id.*id\_text*);  
 latitudeAndLongitudeTextView = findViewById(R.id.*latitude\_Longitude\_text*);  
 addressTextView = findViewById(R.id.*address\_text*);  
  
 id = "Id : ";  
 latitudeAndLongitudeLabel = "Latitude and Longitude : ";  
 addressLabel = "Selected Address : ";  
  
 if (!Places.*isInitialized*()) {  
 Places.*initialize*(getApplicationContext(), *GOOGLE\_PLACES\_API\_KEY*);  
 }  
  
 *// Initialize the AutocompleteSupportFragment.* AutocompleteSupportFragment autocompleteFragment = (AutocompleteSupportFragment) getSupportFragmentManager().findFragmentById(R.id.*autocomplete\_fragment*);  
  
 *// Specify the types of place data to return.* assert autocompleteFragment != null;  
 List<Place.Field> places = new ArrayList<>();  
 places.add(Place.Field.*ID*);  
 places.add(Place.Field.*NAME*);  
 places.add(Place.Field.*ADDRESS*);  
 places.add(Place.Field.*LAT\_LNG*);  
 autocompleteFragment.setPlaceFields(places);  
  
 *// Set up a PlaceSelectionListener to handle the response.* autocompleteFragment.setOnPlaceSelectedListener(new PlaceSelectionListener() {  
 @Override  
 public void onPlaceSelected(Place place) {  
 *// TODO: Get info about the selected place.* idTextView.setText(id+place.getId());  
 latitudeAndLongitudeTextView.setText(latitudeAndLongitudeLabel+String.*valueOf*(place.getLatLng()));  
 addressTextView.setText(addressLabel+place.getName());  
 addMarker(place);  
 }  
  
 @Override  
 public void onError(Status status) {  
 *// TODO: Handle the error.* Log.*i*(TAG, "An error occurred: " + status);  
 }  
 });  
  
 }  
  
 @Override  
 public void onMapReady(GoogleMap map) {  
 googleMap = map;  
  
 *// Add a marker in Sydney and move the camera* LatLng sydney = new LatLng(-34, 151);  
 googleMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));  
 googleMap.moveCamera(CameraUpdateFactory.*newLatLng*(sydney));  
  
 googleMap.setOnMapClickListener(new GoogleMap.OnMapClickListener() {  
 @Override  
 public void onMapClick(LatLng latLng) {  
 Toast.*makeText*(MainActivity.this,latLng+"",Toast.*LENGTH\_LONG*).show();  
 }  
 });  
 }  
  
 public void addMarker(Place p){  
  
 MarkerOptions markerOptions = new MarkerOptions();  
  
 markerOptions.position(p.getLatLng());  
 markerOptions.title(p.getName()+"");  
 markerOptions.icon(BitmapDescriptorFactory.*defaultMarker*(BitmapDescriptorFactory.*HUE\_AZURE*));  
  
 googleMap.addMarker(markerOptions);  
 googleMap.moveCamera(CameraUpdateFactory.*newLatLng*(p.getLatLng()));  
 googleMap.animateCamera(CameraUpdateFactory.*zoomTo*(13));  
 }  
}









**PLACE DETAILS**

Places SDK can be requested to provide extra information about the place. When Place request is constructed you can provide what type of data to be returned. As in the example above we have added LatLon and Name in the fields to be returned.

List<Place.Field> fields = Arrays.asList(Place.Field.***ID***, Place.Field.***NAME***, Place.Field.***LAT\_LNG***);

In addition to these fields we can request following fields :

Place.Field.ADDRESS

Place.Field.ID

Place.Field.LAT\_LNG

Place.Field.NAME

Place.Field.OPENING\_HOURS

Place.Field.PHONE\_NUMBER

Place.Field.PHOTO\_METADATAS

Place.Field.PLUS\_CODE

Place.Field.PRICE\_LEVEL

Place.Field.RATING

Place.Field.TYPES

Place.Field.USER\_RATINGS\_TOTAL

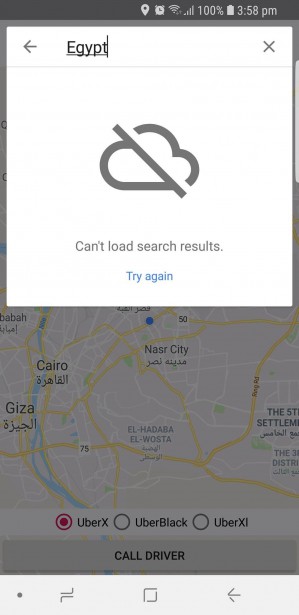
Place.Field.VIEWPORT

Place.Field.WEBSITE\_URI

***If you have any query or suggestion do comment below. Happy coding!***

When I try to search at Place Autocomplete I got Can't Load Search Results and the logs say "Error while autocompleting : OVER\_QUERY\_LIMIT"

I have enabled <https://console.cloud.google.com/> and API key works well.

[](https://i.stack.imgur.com/NHeHj.jpg)

You are getting OVER\_QUERY\_LIMIT message because you have not enabled billing for your project in [developers console](https://console.developers.google.com).

To use the Places SDK for Android, you must include an [API key](https://developers.google.com/places/android-sdk/signup) with all API requests and you must [enable billing](https://console.cloud.google.com/project/_/billing/enable?redirect=https://developers.google.com/places/android-sdk/usage-and-billing?dialogOnLoad=billing-enabled&_ga=2.182998947.461830153.1550562032-1721692727.1545130792) on each of your projects.

**SKU: Basic Data**

Fields in the Basic category are included in the base cost of the Places request, and do not result in any additional charge. The Basic Data SKU is triggered when any of these fields are requested: ADDRESS, ID, LAT\_LNG, NAME, OPENING\_HOURS, PHOTO\_METADATAS, PLUS\_CODE, TYPES, USER\_RATINGS\_TOTAL, VIEWPORT.

**EXAMPLE : Custom auto complete search bar In Google Map**

<https://developers.google.com/places/web-service/autocomplete>

<http://learningprogramming.net/mobile/android/search-places-autocomplete-in-google-maps-with-retrofit/>

<https://stackoverflow.com/questions/30350739/autocompletetextview-with-google-places-shown-in-listview-just-like-uber>

<http://codetheory.in/google-place-api-autocomplete-service-in-android-application/>

<https://examples.javacodegeeks.com/android/android-google-places-autocomplete-api-example/>

<http://kvenkataprasad.blogspot.com/2017/03/autocomplete-google-places-api-example.html>

<https://github.com/manshachuttani/CustomPlaceAutocomplete>

<https://techstricks.com/custom-google-place-autocomplete-android/>

<http://yasirameen.com/2017/08/google-places-api-for-android/>