**Write a program to search a specific location on Google Map**

**activity\_maps.xml code**

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:map="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:name="com.google.android.gms.maps.SupportMapFragment"

    android:layout\_width="match\_parent"

    android:layout\_height="match\_parent"

    tools:context="net.locationsearchexample.MapsActivity">

    <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" />

    <fragment xmlns:android="http://schemas.android.com/apk/res/android"

        xmlns:map="http://schemas.android.com/apk/res-auto"

        xmlns:tools="http://schemas.android.com/tools"

        android:id="@+id/map"

        android:name="com.google.android.gms.maps.SupportMapFragment"

        android:layout\_width="match\_parent"

        android:layout\_height="match\_parent"

        android:layout\_below="@id/place\_autocomplete\_fragment" />

</RelativeLayout>

**AndroidManifest.xml code:**

<?xml version="1.0" encoding="utf-8"?>

<manifest package="com.GoogleMaps.googlemaps"

xmlns:android="http://schemas.android.com/apk/res/android">

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>

<uses-permission android:name="android.permission.INTERNET"/>

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<meta-data

android:name="com.google.android.geo.API\_KEY"

android:value="@string/google\_maps\_key"/>

<activity

android:name="com.abhiandroid.GoogleMaps.googlemaps.MapsActivity"

android:label="@string/title\_activity\_maps">

<intent-filter>

<action android:name="android.intent.action.MAIN"/>

<category android:name="android.intent.category.LAUNCHER"/>

</intent-filter>

</activity>

</application>

</manifest>

MapsActivity.java

package net.locationsearchexample;

import android.support.v4.app.FragmentActivity;

import android.os.Bundle;

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 MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity\_maps);

        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

                .findFragmentById(R.id.map);

        mapFragment.getMapAsync(this);

    }

  PlaceAutocompleteFragment autocompleteFragment = (PlaceAutocompleteFragment)

                getFragmentManager().findFragmentById(R.id.place\_autocomplete\_fragment);

        autocompleteFragment.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));

            }

            @Override

            public void onError(Status status) {

            }

        });

    @Override

    public void onMapReady(GoogleMap googleMap) {

        mMap = googleMap;

        LatLng sydney = new LatLng(-34, 151);

        mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));

        mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));

    }

}

**Write a program to calculate distance between two locations on Google Map.**

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout\_width="match\_parent"

    android:layout\_height="match\_parent"

    tools:context=".MapsActivity">

    <fragment xmlns:android="http://schemas.android.com/apk/res/android"

        xmlns:map="http://schemas.android.com/apk/res-auto"

        xmlns:tools="http://schemas.android.com/tools"

        android:id="@+id/map"

        android:name="com.google.android.gms.maps.SupportMapFragment"

        android:layout\_width="match\_parent"

        android:layout\_height="match\_parent"

        tools:context="net.simplifiedcoding.mymapapp.MapsActivity" />

    <LinearLayout

        android:layout\_width="match\_parent"

        android:layout\_height="match\_parent"

        android:gravity="bottom"

        android:orientation="vertical">

        <LinearLayout

            android:layout\_width="match\_parent"

            android:layout\_height="wrap\_content"

            android:background="#cc3b60a7"

            android:orientation="horizontal">

            <Button

                android:id="@+id/buttonSetFrom"

                android:text="Set From"

                android:layout\_width="wrap\_content"

                android:layout\_height="wrap\_content" />

            <Button

                android:id="@+id/buttonSetTo"

                android:text="Set To"

                android:layout\_width="wrap\_content"

                android:layout\_height="wrap\_content" />

            <Button

                android:id="@+id/buttonCalcDistance"

                android:text="Calc Distance"

                android:layout\_width="wrap\_content"

                android:layout\_height="wrap\_content" />

        </LinearLayout>

    </LinearLayout>

</FrameLayout>

**MapsActivity.java**

public class MapsActivity extends FragmentActivity implements

        OnMapReadyCallback,

        GoogleApiClient.ConnectionCallbacks,

        GoogleApiClient.OnConnectionFailedListener,

        GoogleMap.OnMarkerDragListener,

        GoogleMap.OnMapLongClickListener,

        View.OnClickListener{

    //Our Map

    private GoogleMap mMap;

    //To store longitude and latitude from map

    private double longitude;

    private double latitude;

    //From -> the first coordinate from where we need to calculate the distance

    private double fromLongitude;

    private double fromLatitude;

    //To -> the second coordinate to where we need to calculate the distance

    private double toLongitude;

    private double toLatitude;

    //Google ApiClient

    private GoogleApiClient googleApiClient;

    //Our buttons

    private Button buttonSetTo;

    private Button buttonSetFrom;

    private Button buttonCalcDistance;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity\_maps);

        // Obtain the SupportMapFragment and get notified when the map is ready to be used.

        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

                .findFragmentById(R.id.map);

        mapFragment.getMapAsync(this);

        //Initializing googleapi client

        // ATTENTION: This "addApi(AppIndex.API)"was auto-generated to implement the App Indexing API.

        // See https://g.co/AppIndexing/AndroidStudio for more information.

        googleApiClient = new GoogleApiClient.Builder(this)

                .addConnectionCallbacks(this)

                .addOnConnectionFailedListener(this)

                .addApi(LocationServices.API)

                .addApi(AppIndex.API).build();

        buttonSetTo = (Button) findViewById(R.id.buttonSetTo);

        buttonSetFrom = (Button) findViewById(R.id.buttonSetFrom);

        buttonCalcDistance = (Button) findViewById(R.id.buttonCalcDistance);

        buttonSetTo.setOnClickListener(this);

        buttonSetFrom.setOnClickListener(this);

        buttonCalcDistance.setOnClickListener(this);

    }

    @Override

    protected void onStart() {

        googleApiClient.connect();

        super.onStart();

        // ATTENTION: This was auto-generated to implement the App Indexing API.

        // See https://g.co/AppIndexing/AndroidStudio for more information.

        Action viewAction = Action.newAction(

                Action.TYPE\_VIEW, // TODO: choose an action type.

                "Maps Page", // TODO: Define a title for the content shown.

                // TODO: If you have web page content that matches this app activity's content,

                // make sure this auto-generated web page URL is correct.

                // Otherwise, set the URL to null.

                Uri.parse("http://host/path"),

                // TODO: Make sure this auto-generated app deep link URI is correct.

                Uri.parse("android-app://net.simplifiedcoding.googlemapsdistancecalc/http/host/path")

        );

        AppIndex.AppIndexApi.start(googleApiClient, viewAction);

    }

    @Override

    protected void onStop() {

        googleApiClient.disconnect();

        super.onStop();

        // ATTENTION: This was auto-generated to implement the App Indexing API.

        // See https://g.co/AppIndexing/AndroidStudio for more information.

        Action viewAction = Action.newAction(

                Action.TYPE\_VIEW, // TODO: choose an action type.

                "Maps Page", // TODO: Define a title for the content shown.

                // TODO: If you have web page content that matches this app activity's content,

                // make sure this auto-generated web page URL is correct.

                // Otherwise, set the URL to null.

                Uri.parse("http://host/path"),

                // TODO: Make sure this auto-generated app deep link URI is correct.

                Uri.parse("android-app://net.simplifiedcoding.googlemapsdistancecalc/http/host/path")

        );

        AppIndex.AppIndexApi.end(googleApiClient, viewAction);

    }

    //Getting current location

    private void getCurrentLocation() {

        mMap.clear();

        //Creating a location object

        Location location = LocationServices.FusedLocationApi.getLastLocation(googleApiClient);

        if (location != null) {

            //Getting longitude and latitude

            longitude = location.getLongitude();

            latitude = location.getLatitude();

            //moving the map to location

            moveMap();

        }

    }

    //Function to move the map

    private void moveMap() {

        //Creating a LatLng Object to store Coordinates

        LatLng latLng = new LatLng(latitude, longitude);

        //Adding marker to map

        mMap.addMarker(new MarkerOptions()

                .position(latLng) //setting position

                .draggable(true) //Making the marker draggable

                .title("Current Location")); //Adding a title

        //Moving the camera

        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));

        //Animating the camera

        mMap.animateCamera(CameraUpdateFactory.zoomTo(15));

    }

    @Override

    public void onMapReady(GoogleMap googleMap) {

        mMap = googleMap;

        LatLng latLng = new LatLng(-34, 151);

        mMap.addMarker(new MarkerOptions().position(latLng).draggable(true));

        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));

        mMap.setOnMarkerDragListener(this);

        mMap.setOnMapLongClickListener(this);

    }

    @Override

    public void onConnected(Bundle bundle) {

        getCurrentLocation();

    }

    @Override

    public void onConnectionSuspended(int i) {

    }

    @Override

    public void onConnectionFailed(ConnectionResult connectionResult) {

    }

    @Override

    public void onMapLongClick(LatLng latLng) {

        //Clearing all the markers

        mMap.clear();

        //Adding a new marker to the current pressed position

        mMap.addMarker(new MarkerOptions()

                .position(latLng)

                .draggable(true));

        latitude = latLng.latitude;

        longitude = latLng.longitude;

    }

    @Override

    public void onMarkerDragStart(Marker marker) {

    }

    @Override

    public void onMarkerDrag(Marker marker) {

    }

    @Override

    public void onMarkerDragEnd(Marker marker) {

        //Getting the coordinates

        latitude = marker.getPosition().latitude;

        longitude = marker.getPosition().longitude;

        //Moving the map

        moveMap();

    }

    @Override

    public void onClick(View v) {

        if(v == buttonSetFrom){

            fromLatitude = latitude;

            fromLongitude = longitude;

            Toast.makeText(this,"From set",Toast.LENGTH\_SHORT).show();

        }

        if(v == buttonSetTo){

            toLatitude = latitude;

            toLongitude = longitude;

            Toast.makeText(this,"To set",Toast.LENGTH\_SHORT).show();

        }

        if(v == buttonCalcDistance){

            //This method will show the distance and will also draw the path

            calculateDistance();

        }

    }

}