**Practical No. 8**

**Title: Android program to work with google maps and locations**

**Aim: Create an application to demonstrate google maps and locations**

# Exercise - Create android application to demonstrate google maps and locations

**Implementation: Program:**

# MainActivity.java

package com.example.googlemap;

import androidx.fragment.app.FragmentActivity;

import android.os.Bundle; import android.view.View;

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.internal.ICameraUpdateFactoryDelegate; import com.google.android.gms.maps.model.LatLng; import com.google.android.gms.maps.model.MarkerOptions; import com.example.googlemap.databinding.ActivityMapsBinding;

public class MapsActivity extends FragmentActivity implements

OnMapReadyCallback {

private GoogleMap mMap;

private ActivityMapsBinding binding;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState);

binding = ActivityMapsBinding.inflate(getLayoutInflater()); setContentView(binding.getRoot());

// 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);

}

/\*\*

* Manipulates the map once available.
* This callback is triggered when the map is ready to be used. \* This is where we can add markers or lines, add listeners or move the camera. In this case,
* we just add a marker near Sydney, Australia.
* If Google Play services is not installed on the device, the user will be prompted to install
* it inside the SupportMapFragment. This method will only be triggered once the user has
* installed Google Play services and returned to the app.

\*/ @Override

public void onMapReady(GoogleMap googleMap) { mMap = googleMap;

// Add a marker in Sydney and move the camera LatLng mumbai = new LatLng( 16.992500,73.294197 );

mMap.addMarker(new MarkerOptions().position(mumbai).title("Marker in Mumbai"));

mMap.moveCamera(CameraUpdateFactory.newLatLng(mumbai)); mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(mumbai,12)); mMap.getUiSettings().setZoomControlsEnabled(true);

}

public void onZoom(View view)

{

if (view.getId()==R.id.button){

mMap.animateCamera(CameraUpdateFactory.zoomIn());

}

if (view.getId()==R.id.button2){

mMap.animateCamera(CameraUpdateFactory.zoomOut());

}

}

}

# activity\_main.xml

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

<LinearLayout 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:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical">

<fragment android:id="@+id/map"

android:name="com.google.android.gms.maps.SupportMapFragment" android:layout\_width="match\_parent" android:layout\_height="598dp" tools:context=".MapsActivity" />

<LinearLayout

android:layout\_width="match\_parent" android:layout\_height="92dp" android:orientation="horizontal">

<Button

android:id="@+id/button" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_weight="1" android:background="#CD2828" android:onClick="onZoom" android:text="ZoomIn" android:textSize="20sp" />

<Button

android:id="@+id/button2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_weight="1" android:background="#A32929" android:onClick="onZoom" android:text="ZoomOut" android:textSize="20sp" />

</LinearLayout>

</LinearLayout>

**Output:** 