Practical No: 01

Display Google map

MapsActivity.java:

```
package com.example.gmap;
import androidx.fragment.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;
import com.example.gmap_24.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(19.0760, 72.8777);
    mMap.addMarker(new MarkerOptions().position(mumbai).title("Marker in
Mumbai"));
    mMap.moveCamera(CameraUpdateFactory.newLatLng(mumbai));
  }
}
```

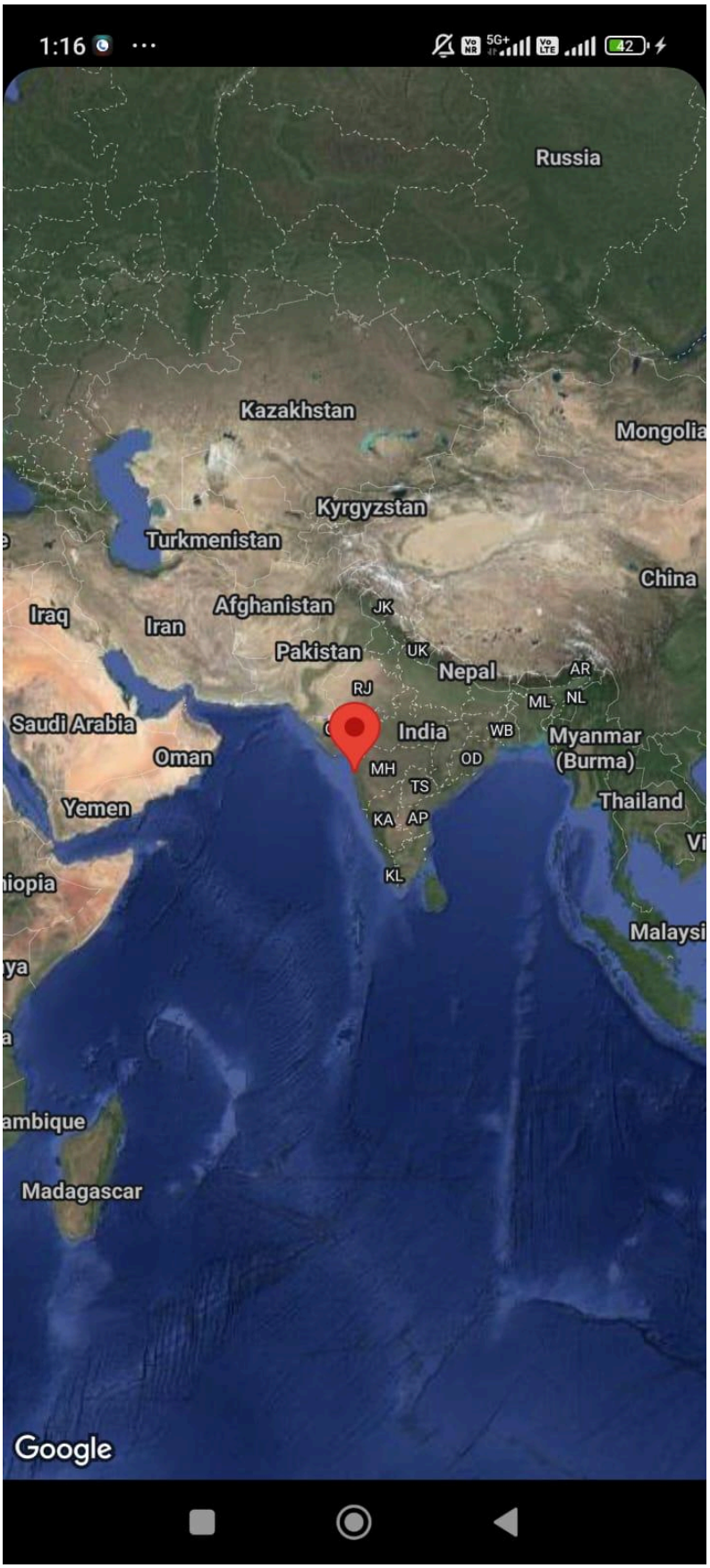
AndroidManifest.xml:

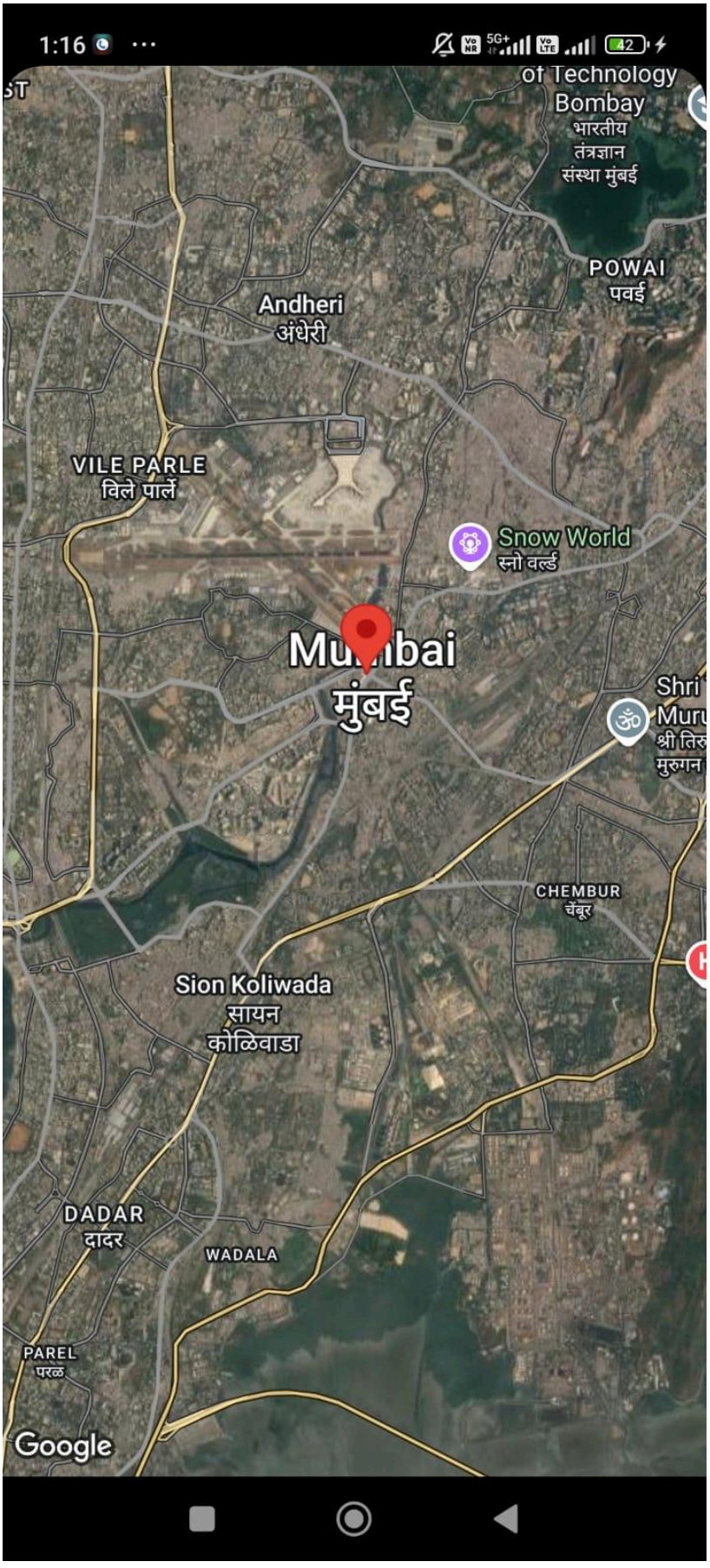
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  package="com.example.gmap_24">
  <application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.GMap_24"
    tools:targetApi="31">
    <!--
      TODO: Before you run your application, you need a Google Maps API key.
      To get one, follow the directions here:

https://developers.google.com/maps/documentation/android-sdk/get-api-key
      Once you have your API key (it starts with "AIza"), define a new property
in your
```

```
        project's local.properties file (e.g. MAPS_API_KEY=Aiza...), and replace
the
        "YOUR_API_KEY" string in this file with "${MAPS_API_KEY}".
    -->
    <meta-data
      android:name="com.google.android.geo.API_KEY"
      android:value="AIzaSyAYO9Z1hQ6bpXjnrYvHrxqFygty66s1BgY" />
    <activity
      android:name=".MapsActivity"
      android:exported="true"
      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>
```

Output:


1:16
Russia
Kazakhstan
Mongolia
Kyrgyzstan
Turkmenistan
China
Afghanistan
JK
Iraq
Iran
Pakistan
UK
Nepal
AR
RJ
ML
NL
Saudi Arabia
India
WB
Myanmar
(Burma)
Oman
MH
OD
TS
Thailand
Yemen
KA
AP
KL
Malaysi
ambique
Madagascar
Google


1:16
of Technology
Bombay
भारतीय
तंत्रज्ञान
संस्था मुंबई
POWAI
पवई
Andheri
अंधेरी
VILE PARLE
विले पार्ले
Snow World
स्नो वर्ल्ड
Mumbai
मुंबई
CHEMBUR
चेंबूर
Sion Koliwada
सायन
कोळिवाडा
DADAR
दादर
WADALA
PAREL
परळ
Google

Practical No: 02

Create an android application to display current location of device (latitude longitude & address)

MainActivity.java:

```
package com.example.geocoder;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.tasks.OnSuccessListener;
import android.Manifest;
import android.annotation.SuppressLint;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;
import java.io.IOException;
import java.util.List;
import java.util.Locale;
public class MapsActivity extends AppCompatActivity {
  private static final int LOCATION_PERMISSION_REQUEST_CODE = 1;
  FusedLocationProviderClient fusedLocationClient;
  private TextView latitudeTextView;
  private TextView longitudeTextView;
  private TextView addressTextView;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);
    latitudeTextView = findViewById(R.id.latitudeTextView);
    longitudeTextView = findViewById(R.id.longitudeTextView);
    addressTextView = findViewById(R.id.addressTextView);
```

```
    fusedLocationClient =
LocationServices.getFusedLocationProviderClient(this);
    // Check and request location permission if not granted.
    if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
      ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION},
LOCATION_PERMISSION_REQUEST_CODE);
    } else {
      // Permission is already granted, so request the location.
      requestLocation();
    }
  }
  private void requestLocation() {
    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED)
      if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        return;
      }
    fusedLocationClient.getLastLocation()
        .addOnSuccessListener(this, new OnSuccessListener<Location>() {
          @SuppressLint("SetTextI18n")
          @Override
          public void onSuccess(Location location) {
            if (location != null) {
              double latitude = location.getLatitude();
              double longitude = location.getLongitude();
              // Display latitude and longitude
              latitudeTextView.setText("Latitude: " + latitude);
              longitudeTextView.setText("Longitude: " + longitude);
              // Get detailed address
              getAddressFromLocation(location);
            } else {
```

```
            Toast.makeText(MapsActivity.this, "Location not available",
Toast.LENGTH_SHORT).show();
          }
        }
        @SuppressLint("SetTextI18n")
        private void getAddressFromLocation(Location location) {
          try {
            Geocoder geocoder = new
Geocoder(MapsActivity.this,Locale.getDefault());
            List<Address> addresses =
geocoder.getFromLocation(location.getLatitude(), location.getLongitude(), 4);
            if (!addresses.isEmpty()) {
              addressTextView.setText(addressTextView.getText() +
"\n"+addresses.get(0).getAddressLine(0)+", "+
                  addresses.get(0).getAddressLine(1)+",
"+addresses.get(0).getAddressLine(2));
              //String address = addresses.get(0).getAddressLine(0); // Get the first
address
              // addressTextView.setText("Address: " + address);
            } else {
              addressTextView.setText("Address not found");
            }
          } catch (IOException e) {
            e.printStackTrace();
            addressTextView.setText("Error getting address");
          }
        }
      });
  }
  @Override
  public void onRequestPermissionsResult( int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == LOCATION_PERMISSION_REQUEST_CODE) {
      if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
        // Permission granted, request the location.
```

```
        requestLocation();
      } else {
        Toast.makeText(this, "Location permission is required to use this feature",
Toast.LENGTH_SHORT).show();
      }
    }
  }
}
```

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="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"
  android:padding="16dp"
  tools:context=".MapsActivity">
  <TextView
    android:id="@+id/titleTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Location Information"
    android:textSize="24sp"
    android:textStyle="bold"
    android:layout_marginBottom="16dp" />
  <TextView
    android:id="@+id/latitudeTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Latitude: "
    android:textSize="18sp"
    android:textStyle="bold"
    android:layout_marginBottom="8dp" />
  <TextView
    android:id="@+id/longitudeTextView"
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Longitude: "
    android:textSize="18sp"
    android:textStyle="bold"
    android:layout_marginBottom="8dp" />
  <TextView
    android:id="@+id/addressTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Address: "
    android:textSize="18sp"
    android:textStyle="bold"
    android:textColor="@android:color/holo_blue_dark" />
</LinearLayout>
```

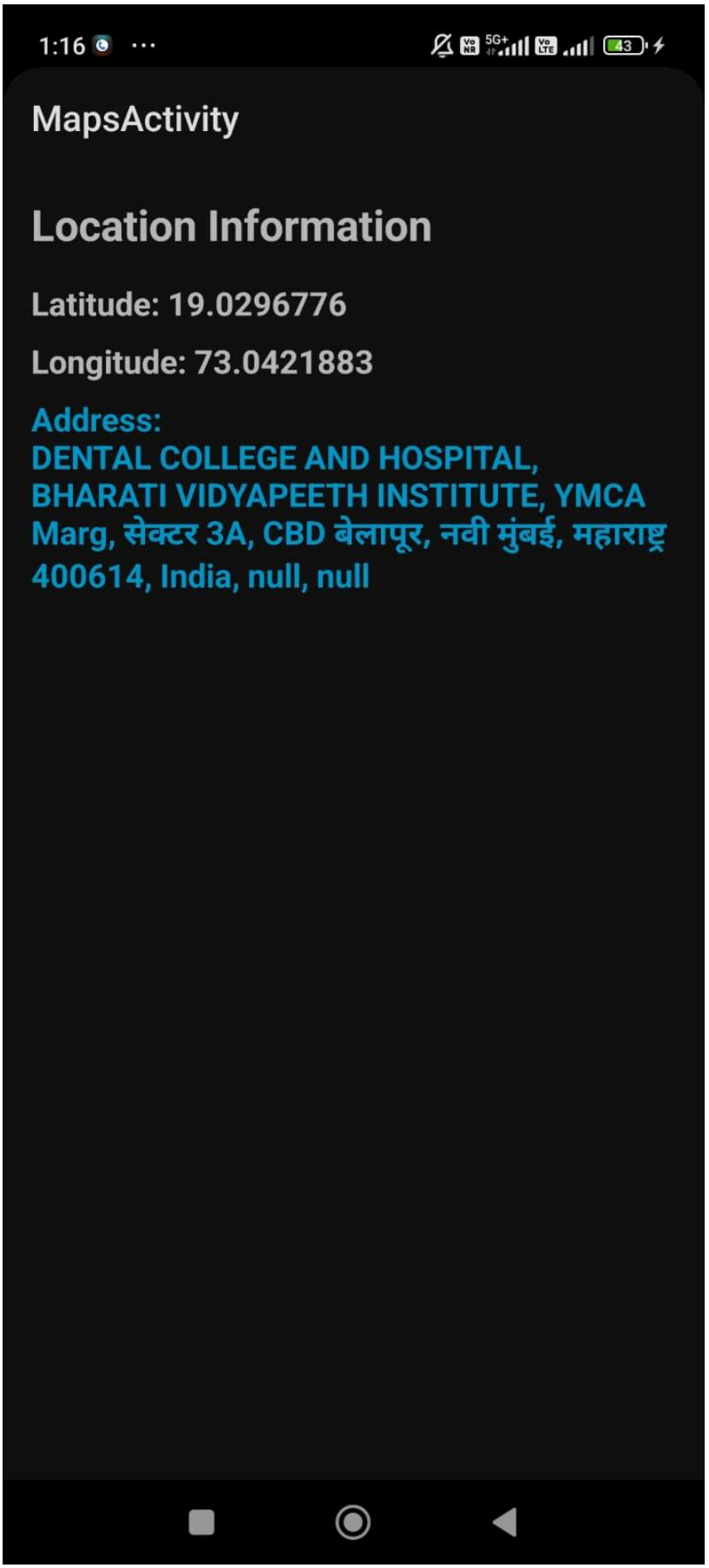
Androidmanifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  package="com.example.geocoder_24">
  <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
  <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
  <uses-permission android:name="android.permission.INTERNET"/>
  <application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.Geocoder_24"
    tools:targetApi="31">
    <!--
```

```
        TODO: Before you run your application, you need a Google Maps API key.
        To get one, follow the directions here:

https://developers.google.com/maps/documentation/android-sdk/get-api-key
        Once you have your API key (it starts with "AIza"), define a new property
in your
        project's local.properties file (e.g. MAPS_API_KEY=Aiza...), and replace
the
        "YOUR_API_KEY" string in this file with "${MAPS_API_KEY}".
    -->
    <meta-data
      android:name="com.google.android.geo.API_KEY"
      android:value="AIzaSyBMaKf_0Cilwq4hFQ__QP7j9tmkdi8SCm4" />
    <activity
      android:name=".MapsActivity"
      android:exported="true"
      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>
```

Output:


1:16
MapsActivity
Location Information
Latitude: 19.0296776
Longitude: 73.0421883
Address:
DENTAL COLLEGE AND HOSPITAL,
BHARATI VIDYAPEETH INSTITUTE, YMCA
Marg, सेक्टर 3A, CBD बेलापूर, नवी मुंबई, महाराष्ट्र
400614, India, null, null