Android Practical (209) – Assignment-4

Question-1

1. Write an android application

1) to send message (sms).
2) to receive alert for incoming message using broadcast receiver
3) to read and display the messages in message box.

----------AndroidManifest.xml----------

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.question_1">

    <uses-permission android:name="android.permission.READ_PHONE_STATE" /> <!-- <uses-permission android:name="android.permission.READ_PRECISE_PHONE_STATE"/> -->
    <uses-permission android:name="android.permission.READ_SMS" />
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.RECEIVE_SMS" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Question1">
        <activity
            android:name=".Notification_Res_Activity"
            android:exported="true" />
        <activity
            android:name=".receiveSMSActivity"
            android:exported="true" />
        <activity
            android:name=".sendSMSActivity"
            android:exported="true" />

        <receiver
```

```
        android:name=".MyReceiver"
        android:enabled="true"
        android:exported="true">
        <intent-filter>

          <!--
          <action android:name="android.intent.action.BATTERY_LOW" />
          <action android:name="android.intent.action.CAMERA_BUTTON" />
          -->
          <action android:name="android.provider.Telephony.SMS_RECEIVED" />
        </intent-filter>
      </receiver>

      <activity
        android:name=".MainActivity"
        android:exported="true">
        <intent-filter>
          <action android:name="android.intent.action.MAIN" />

          <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
      </activity>
    </application>

</manifest>
```

========== java files==========

----------MainActivity.java----------

```
package com.example.question_1;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.app.NotificationManagerCompat;
import androidx.core.content.ContextCompat;

import android.Manifest;
import android.annotation.SuppressLint;
import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.os.Build;
import android.os.Bundle;
import android.telecom.TelecomManager;
import android.telephony.TelephonyManager;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    String NotifyChannelID = "NotificationChannel1001";
    int Notification_Req_Code = 1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);



        //Dynamic Rgistration of Notification using Broadcast Receiver
/*
        IntentFilter filter = new IntentFilter(android.provider.Telephony.SMS_RECEIVED);
        MyReceiver mr = new MyReceiver();
        registerReceiver(mr,filter);

*/
       //  getPermision();



    }

    private void getPermision(){
        if(ContextCompat.checkSelfPermission(this,
                Manifest.permission.READ_PHONE_STATE) == PackageManager.PERMISSION_GRANTED){

            showPhoneState();
        }
        else{

            //Dynamic Permission

            String[] permissions = new String[]{Manifest.permission.READ_PHONE_STATE};
            ActivityCompat.requestPermissions(this,permissions,1);
        }
    }

    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[]
grantResults) {
```

```
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if(requestCode == 1){
      if(grantResults.length >0){
        if(grantResults[0] == PackageManager.PERMISSION_GRANTED){
          showPhoneState();
        }
        else{
          Toast.makeText(getApplicationContext(), "Permission Not Granted.....", Toast.LENGTH_SHORT).show();
        }
      }
    }
  }

  private void showPhoneState(){
    TelephonyManager tm = (TelephonyManager) getSystemService(Context.TELEPHONY_SERVICE);

    String networkCountryISO = tm.getNetworkCountryIso();
    String SIMCountryISO = tm.getSimCountryIso();

    String softwareVersion =  null;
   //  String EMI = tm.getEmi();

    Toast.makeText(getApplicationContext(), networkCountryISO+"\n"+SIMCountryISO+"\n",
Toast.LENGTH_SHORT).show();

  }

  public void onbtnCallBroadClick(View view) {

    getPermision();
    /*
    Intent intent = new Intent();
    intent.setAction("com.example.question_2");
    sendBroadcast(intent);
```

```
    */
}

public void onbtnSend(View view) {
    Intent i = new Intent(this,sendSMSActivity.class);
    startActivity(i);
}

public void onbtnReceive(View view) {
    Intent i = new Intent(this,receiveSMSActivity.class);
    startActivity(i);
}


public void sendNotification(){
    Notification.Builder builder;

    if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
        createNotificationChannel();
        builder = new Notification.Builder(this,NotifyChannelID);
        builder.setContentTitle("My Msg Notify");
        builder.setContentText("This is to Notify You...for...Demo....");
        builder.setSmallIcon(R.drawable.ic_launcher_background);

        Notification nn = builder.build();

        NotificationManagerCompat nmc = NotificationManagerCompat.from(this);

        Intent i = new Intent(this,receiveSMSActivity.class);
        PendingIntent pi = PendingIntent.getActivity(this,3,i,PendingIntent.FLAG_UPDATE_CURRENT);
        builder.setContentIntent(pi);

        nmc.notify(Notification_Req_Code,builder.build());
```

```
        }
    }
    public void createNotificationChannel(){
        NotificationChannel nc;
        nc = new NotificationChannel(NotifyChannelID,"Notification Channel 1",
NotificationManager.IMPORTANCE_DEFAULT);
        NotificationManager nm = getSystemService(NotificationManager.class);
        nm.createNotificationChannel(nc);
    }

    public void sendNotificationClick(View view) {
        sendNotification();
    }
}
```

----------sendSMSActivity.java----------

```
package com.example.question_1;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class sendSMSActivity extends AppCompatActivity {

    Button btnSend;
    EditText edtMsg,edtNumber;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_send_smsactivity);

        btnSend = findViewById(R.id.btnSend);
        edtMsg = findViewById(R.id.edtMsg);
        edtNumber = findViewById(R.id.edtNumber);

        btnSend.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                getPermission();
            }
```

```
    });
  }

  private void getPermission(){
    if(ActivityCompat.checkSelfPermission(this,
        Manifest.permission.READ_SMS)== PackageManager.PERMISSION_GRANTED
    && ActivityCompat.checkSelfPermission(this,
        Manifest.permission.SEND_SMS)==PackageManager.PERMISSION_GRANTED){

      sendSMS();

    }
    else{
      String[] permissions = new String[]{Manifest.permission.READ_SMS,Manifest.permission.SEND_SMS};
      ActivityCompat.requestPermissions(this,permissions,1);
    }
  }

  @Override
  public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[]
grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if(requestCode == 1){
      if(grantResults.length > 0){
        if(grantResults[0] == PackageManager.PERMISSION_GRANTED){
          sendSMS();
        }
        else{
          Toast.makeText(getApplicationContext(), "Permission Denied....", Toast.LENGTH_SHORT).show();
        }
      }
    }
  }
```

```
    private void sendSMS(){
        SmsManager sm = SmsManager.getDefault();

        String number = edtNumber.getText().toString();
        String msg = edtMsg.getText().toString();

        sm.sendTextMessage(number,"myPhone",msg,null,null);

        Toast.makeText(getApplicationContext(), "SMS Sent", Toast.LENGTH_SHORT).show();
    }
}
```

----------receiveSMSActivity.java----------

```
package com.example.question_1;

import androidx.appcompat.app.AppCompatActivity;

import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.content.ContentResolver;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.Toast;

import java.util.ArrayList;

public class receiveSMSActivity extends AppCompatActivity {



    Button btnAll,btnSent,btnInbox;
    ListView lstMsg;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_receive_smsactivity);

        lstMsg = findViewById(R.id.lstMsg);
        btnAll = findViewById(R.id.btnAll);
        btnSent = findViewById(R.id.btnSent);
        btnInbox = findViewById(R.id.btnInbox);
```

```
    btnAll.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            refreshSMSAll();
        }
    });

    btnInbox.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            refreshSMSInbox();
        }
    });

    btnSent.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            refreshSMSSent();
        }
    });
}

private void refreshSMSAll(){
    ArrayList<String> smsList = new ArrayList<String>();
    ArrayAdapter aa;

    ContentResolver cr = getContentResolver();
    Uri SMSUri;

    SMSUri = Uri.parse("content://sms");

    Cursor c = cr.query(SMSUri,null,null,null,null);
```

```
    int indexBody,indexAddress,indexStatus;

    indexBody = c.getColumnIndex("body"); //12
    indexAddress = c.getColumnIndex("address"); //2
    indexStatus = c.getColumnIndex("stype"); //-1

    Toast.makeText(getApplicationContext(), indexBody+"  "+indexAddress+" "+indexStatus,
Toast.LENGTH_SHORT).show();

    if(c.getCount() > 0){
      while(c.moveToNext()){
        String str = "";
        String type = (c.getString(0).equals("1"))?"sent":"received";
        str = "SMS Address : "+c.getString(2)+"\nSMS Msg : "+c.getString(12)+"\nSMS Status : "+type;
        smsList.add(str);
      }

      aa = new ArrayAdapter(this,R.layout.list_sms_layout,smsList);

      lstMsg.setAdapter(aa);
    }
    else{
      Toast.makeText(getApplicationContext(), "No SMS ", Toast.LENGTH_SHORT).show();
    }

  }
  private void refreshSMSInbox(){
    ArrayList<String> smsList = new ArrayList<String>();
    ArrayAdapter aa;

    ContentResolver cr = getContentResolver();
    Uri SMSUri;
```

```
    SMSUri = Uri.parse("content://sms/inbox");

    Cursor c = cr.query(SMSUri,null,null,null,null);

    int indexBody,indexAddress,indexStatus;

    indexBody = c.getColumnIndex("body");
    indexAddress = c.getColumnIndex("address");
    indexStatus = c.getColumnIndex("stype");

    while(c.moveToNext()){
       String str = "";
       String type = (c.getString(0).equals("1"))?"sent":"received";
       str = "SMS Address : "+c.getString(2)+"\nSMS Msg : "+c.getString(12)+"\nSMS Status : "+type;
       smsList.add(str);
    }

    aa = new ArrayAdapter(this,R.layout.list_sms_layout,smsList);

    lstMsg.setAdapter(aa);


}
private void refreshSMSSent(){
    ArrayList<String> smsList = new ArrayList<String>();
    ArrayAdapter aa;

    ContentResolver cr = getContentResolver();
    Uri SMSUri;

    SMSUri = Uri.parse("content://sms/sent");

    Cursor c = cr.query(SMSUri,null,null,null,null);
```

```
    int indexBody,indexAddress,indexStatus;

    indexBody = c.getColumnIndex("body");
    indexAddress = c.getColumnIndex("address");
    indexStatus = c.getColumnIndex("stype");

    while(c.moveToNext()){
       String str = "";
       String type = (c.getString(0).equals("1"))?"sent":"received";
       str = "SMS Address : "+c.getString(2)+"\nSMS Msg : "+c.getString(12)+"\nSMS Status : "+type;
       smsList.add(str);
    }

    aa = new ArrayAdapter(this,R.layout.list_sms_layout,smsList);

    lstMsg.setAdapter(aa);


  }

}
```

----------MyReceiver.java----------Broadcast_Receiver----------

```
package com.example.question_1;

import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.telephony.SmsMessage;
import android.widget.Toast;

import androidx.core.app.NotificationManagerCompat;

public class MyReceiver extends BroadcastReceiver {

   String NotifyChannelID = "NotificationChannel1001";
   int Notification_Req_Code = 1;
   @Override
   public void onReceive(Context context, Intent intent) {
      // TODO: This method is called when the BroadcastReceiver is receiving
      // an Intent broadcast.
      //throw new UnsupportedOperationException("Not yet implemented");

      SmsMessage cuurentSMS;
      Bundle intentExtras = intent.getExtras();

      if(intentExtras != null){
         Object[] pdu_objs = (Object[]) intentExtras.get("pdus");
         if(pdu_objs != null){
            //if(Build.VERSION.SDK_INT >= 23){}
```

```
            if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
               String format = intentExtras.getString("format");
               cuurentSMS = SmsMessage.createFromPdu((byte[]) pdu_objs[1],format);
            }
            else{
               cuurentSMS = SmsMessage.createFromPdu((byte[]) pdu_objs[1]);
            }

            String smsBody = cuurentSMS.getMessageBody().toString();
            String address = cuurentSMS.getOriginatingAddress().toString();

            String SMSMsg = "";
            SMSMsg = "SMS Notification : \nAddress: "+address+"\nMsg : "+smsBody;

            Toast.makeText(context.getApplicationContext(), ""+SMSMsg, Toast.LENGTH_SHORT).show();

           sendNotification(SMSMsg,context,intent);
         }
      }

      Toast.makeText(context.getApplicationContext(), "Received Broadcast"+intent.getAction(),
Toast.LENGTH_SHORT).show();
   }

   public void sendNotification(String SMSMsg,Context context,Intent intent) {
      Notification.Builder builder;

      if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
         createNotificationChannel(context,intent);
         builder = new Notification.Builder(context,NotifyChannelID);
         builder.setContentTitle("My REceived MSG NOtyification");
         builder.setContentText(SMSMsg.toString());
         builder.setSmallIcon(R.drawable.ic_launcher_background);
```

```
        Notification nn = builder.build();


        NotificationManagerCompat nmc = NotificationManagerCompat.from(context);


        Intent i = new Intent(Intent.ACTION_VIEW);
        PendingIntent pi = PendingIntent.getActivity(context,3,i,PendingIntent.FLAG_UPDATE_CURRENT);
        builder.setContentIntent(pi);


        nmc.notify(Notification_Req_Code,builder.build());


      }
   }


   public void createNotificationChannel(Context context,Intent intent){
      NotificationChannel nc;
      nc = new NotificationChannel(NotifyChannelID,"Notification Channel 1",
NotificationManager.IMPORTANCE_DEFAULT);
      NotificationManager nm = (NotificationManager)
context.getSystemService(Context.NOTIFICATION_SERVICE);
      nm.createNotificationChannel(nc);
   }
}
```

==========Layout Files==========

----------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"
    tools:context=".MainActivity"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Broadcast Example"/>

    <Button
        android:id="@+id/btnSendNotification"
        android:onClick="sendNotificationClick"
        android:text="Send Notification"
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/btnCallBroad"
        android:onClick="onbtnCallBroadClick"
        android:text="Show Phone STate"
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/btnToSend"
```

```
    android:onClick="onbtnSend"
    android:text="To Send SMS"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>

  <Button
    android:id="@+id/btnToReceive"
    android:onClick="onbtnReceive"
    android:text="To Receive SMS"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>


</LinearLayout>
```

----------activity_send_smsactivity.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"
    tools:context=".sendSMSActivity"
    android:orientation="vertical">

    <EditText
        android:id="@+id/edtNumber"
        android:hint="Enter Mobile Number"
        android:inputType="numberDecimal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <EditText
        android:id="@+id/edtMsg"
        android:hint="Enter Msg"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/btnSend"
        android:text="Send SMS"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

----------activity_receive_smsactivity.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"
    tools:context=".receiveSMSActivity"
    android:orientation="vertical">

    <Button
        android:id="@+id/btnAll"
        android:text="All"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    <Button
        android:id="@+id/btnSent"
        android:text="Sent"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    <Button
        android:id="@+id/btnInbox"
        android:text="Inbox"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    <ListView
        android:id="@+id/lstMsg"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

----------lst_sms_layout.xml----------

```
<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

</TextView>
```

Question-2

2. Write an android application to display current location of your device on Google map.(In Longitude and latititude)

==========AndroisManifest.xml==========

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.question_2">


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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Question2">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />


                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

==========java files==========

----------MainActivity.java----------

```
package com.example.question_2;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.Context;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationManager;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      LocationManager lm;
      String svcName = Context.LOCATION_SERVICE;
      lm = (LocationManager) getSystemService(svcName);

      String provider = LocationManager.GPS_PROVIDER;

      if (ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED
            && ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
```

```
        String[] permissions = new
String[]{Manifest.permission.ACCESS_FINE_LOCATION,Manifest.permission.ACCESS_COARSE_LOCATION};

        ActivityCompat.requestPermissions(this,permissions,101);
        return;
      }

      Location l = lm.getLastKnownLocation(provider);

      updateWithNewLocation(l);
   }

   private void updateWithNewLocation(Location l){
      TextView txtLocation;
      txtLocation = findViewById(R.id.txtLocation);

      String latiLongiString = "No Location Found";
      if(l != null){
         double lati = l.getLatitude();
         double longi = l.getLongitude();

         latiLongiString = "Latitude : "+lati+"\nLongitude : "+longi;
      }
      else{
         Toast.makeText(getApplicationContext(), "No Locatio0n Found", Toast.LENGTH_SHORT).show();
      }

      txtLocation.setText("Your Current Location is :\n"+latiLongiString);
   }
}
```

==========layout files==========

----------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"
    tools:context=".MainActivity"
    android:orientation="vertical">

    <TextView
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Current Location"
        android:textSize="30sp"/>

    <TextView
        android:id="@+id/txtLocation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>


</LinearLayout>
```

Question-2(Map Activity)

2. Write an android application to display current location of your device on **Google map**

==========AndroisManifest.xml==========

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.current_location">

    <!--
        The ACCESS_COARSE/FINE_LOCATION permissions are not required to use
        Google Maps Android API v2, but you must specify either coarse or fine
        location permissions for the "MyLocation" functionality.
    -->
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Current_Location">

        <!--
            The API key for Google Maps-based APIs is defined as a string resource.
            (See the file "res/values/google_maps_api.xml").
            Note that the API key is linked to the encryption key used to sign the APK.
            You need a different API key for each encryption key, including the release key that is used to
            sign the APK for publishing.
            You can define the keys for the debug and release targets in src/debug/ and src/release/.
        -->
```

```
        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />

        <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>
```

==========java files==========

----------MapsActivity.java----------

```
package com.example.current_location;

import androidx.annotation.NonNull;
import androidx.core.app.ActivityCompat;
import androidx.fragment.app.FragmentActivity;

import android.Manifest;
import android.content.pm.PackageManager;
import android.icu.util.ICUUncheckedIOException;
import android.location.Location;
import android.os.Bundle;
import android.widget.Toast;

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
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.current_location.databinding.ActivityMapsBinding;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;
    private ActivityMapsBinding binding;

    //
    Location currentLocation;
```

```
    FusedLocationProviderClient fusedLocationProviderClient;
    private static final int REQUEST_CODE = 101;
    //



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

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

        //
        fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);
        getCurrentLocation();
        //
/*
        // 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 surat = new LatLng(currentLocation.getLatitude(), currentLocation.getLongitude());
    mMap.addMarker(new MarkerOptions().position(surat).title("Marker on Current Location"));
    mMap.moveCamera(CameraUpdateFactory.newLatLng(surat));
}

//
private void getCurrentLocation(){
    if(ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED
    && ActivityCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED){

        String[] permissions = new
String[]{Manifest.permission.ACCESS_COARSE_LOCATION,Manifest.permission.ACCESS_FINE_LOCATION};
        ActivityCompat.requestPermissions(this,permissions,REQUEST_CODE);

        return;

    }

    Task<Location> task = fusedLocationProviderClient.getLastLocation();
    task.addOnSuccessListener(new OnSuccessListener<Location>() {
        @Override
        public void onSuccess(Location location) {
            if(location != null){
                currentLocation = location;
                Toast.makeText(getApplicationContext(),
currentLocation.getLatitude()+"\n"+currentLocation.getLongitude(), Toast.LENGTH_LONG).show();
```

```
            SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map);

            mapFragment.getMapAsync(MapsActivity.this);

          }

        }

    });

  }


  @Override

  public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[]
grantResults) {

    switch (REQUEST_CODE){

      case REQUEST_CODE:

        if(grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED){

          getCurrentLocation();

        }

        break;

    }


    super.onRequestPermissionsResult(requestCode, permissions, grantResults);

  }


  //

}
```

==========layout files==========

----------activity_maps.xml----------

```
<?xml version="1.0" encoding="utf-8"?>
<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=".MapsActivity" />
```

==========Gradle files==========

----------build.gradle Module----------

```
plugins {
    id 'com.android.application'
}

android {
    compileSdk 31

    defaultConfig {
        applicationId "com.example.current_location"
        minSdk 28
        targetSdk 31
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
    buildFeatures {
        viewBinding true
    }
}
```

```
dependencies {

    implementation 'androidx.appcompat:appcompat:1.4.1'
    implementation 'com.google.android.material:material:1.6.0'
    implementation 'com.google.android.gms:play-services-maps:18.0.2'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
    implementation 'com.google.android.gms:play-services-location:19.0.1'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
}
```

Question-3

3. Write an android application to demonstrate the use of geocoding and reverse geocoding

==========AndroidManifest.xml==========

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.question_3">

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Question3">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

==========java files==========

==========MainActivity.java==========

```
package com.example.question_3;

import androidx.appcompat.app.AppCompatActivity;

import android.location.Address;
import android.location.Geocoder;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    Button btnGet,btnSet;
    EditText edtAddress,edtLatitude,edtLongitude;
    TextView txtAddress,txtLatitude,txtLongitude;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btnGet = findViewById(R.id.btnGet);
        btnSet = findViewById(R.id.btnSet);
        edtAddress = findViewById(R.id.edtAddress);
        edtLatitude = findViewById(R.id.edtLatitude);
```

```
edtLongitude = findViewById(R.id.edtLongitde);
txtAddress = findViewById(R.id.txtAddress);
txtLatitude = findViewById(R.id.txtLatitude);
txtLongitude = findViewById(R.id.txtLongitude);


btnGet.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View view) {
      String strAdd ;
      strAdd = edtAddress.getText().toString();

      Geocoder gc = new Geocoder(getApplicationContext());
      if(gc.isPresent()){
         List<Address> list = null;
         try {
            list = gc.getFromLocationName(strAdd,1);
         } catch (IOException e) {
            e.printStackTrace();
         }

         if(list != null){
            Address address = list.get(0);
            double latitude = address.getLatitude();
            double longitude = address.getLongitude();

            txtLatitude.setText(""+latitude);
            txtLongitude.setText(""+longitude);
         }
         else{
            Toast.makeText(getApplicationContext(), "Address Not Found", Toast.LENGTH_SHORT).show();
         }
      }
      else{
```

```
            Toast.makeText(getApplicationContext(), "GeoCoder Not present", Toast.LENGTH_SHORT).show();
        }
    }
});

btnSet.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        double strLati,strLongi;
        strLati = Float.parseFloat(edtLatitude.getText().toString());
        strLongi = Float.parseFloat(edtLongitude.getText().toString());

        Geocoder gc = new Geocoder(getApplicationContext());
        List<Address> list = null;
        if(gc.isPresent()){
            try {
                list = gc.getFromLocation(strLati,strLongi,1);
            } catch (IOException e) {
                e.printStackTrace();
            }

            if(list != null){
                Address address = list.get(0);
                StringBuffer str = new StringBuffer();
                str.append("Locality : "+address.getLocality()+"\n");
                str.append("Country : "+address.getCountryName());

                txtAddress.setText(str);
            }
            else{
                Toast.makeText(getApplicationContext(), "Address Not Found", Toast.LENGTH_SHORT).show();
            }
        }
        else{
```

```
                Toast.makeText(getApplicationContext(), "GeoCoder Not present", Toast.LENGTH_SHORT).show();
            }
        }
    });
  }
}
```

==========**Layout** files==========

==========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"
    tools:context=".MainActivity"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textSize="30sp"
        android:text="Geocoding" />

    <EditText
        android:id="@+id/edtAddress"
        android:hint="Enter Address"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <TextView
        android:id="@+id/txtLatitude"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <TextView
        android:id="@+id/txtLongitude"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <Button
```

```
    android:id="@+id/btnGet"
    android:text="Get"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>



<TextView
    android:text="Reverse Geocoding"
    android:gravity="center"
    android:textSize="30sp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>

<EditText
    android:id="@+id/edtLatitude"
    android:hint="Latitude"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>

<EditText
    android:id="@+id/edtLongitde"
    android:hint="Logitude"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>

<TextView
    android:id="@+id/txtAddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"/>

<Button
    android:id="@+id/btnSet"
```

```
        android:text="Set"
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

</LinearLayout>
```