**Experiment - 8**

**Name : Ansari Ushair Roll no : 14DCO53**

**Class : BE.CO Batch : 03**

**Aim : Develop a native application that uses GPS location information.**

**#Theory**

**Program:**

**MyActivity.java**

package com.example.exp8;

import android.app.Activity;

import android.app.AlertDialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.location.Location;

import android.location.LocationManager;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.provider.Settings;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MyActivity extends Activity {

Button btnGPSShowLocation;

Button btnShowAddress;

TextView tvAddress;

AppLocationService appLocationService;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

tvAddress = (TextView) findViewById(R.id.tvAddress);

appLocationService = new AppLocationService(

MyActivity.this);

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

btnGPSShowLocation.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View arg0) {

Location gpsLocation = appLocationService

.getLocation(LocationManager.GPS\_PROVIDER);

if (gpsLocation != null) {

double latitude = gpsLocation.getLatitude();

double longitude = gpsLocation.getLongitude();

String result = "Latitude: " + gpsLocation.getLatitude() +

" Longitude: " + gpsLocation.getLongitude();

tvAddress.setText(result);

} else {

showSettingsAlert();

}

}

});

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

btnShowAddress.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View arg0) {

Location location = appLocationService

.getLocation(LocationManager.GPS\_PROVIDER);

//you can hard-code the lat & long if you have issues with getting it

//remove the below if-condition and use the following couple of lines

//double latitude = 37.422005;

//double longitude = -122.084095

if (location != null) {

double latitude = location.getLatitude();

double longitude = location.getLongitude();

LocationAddress locationAddress = new LocationAddress();

locationAddress.getAddressFromLocation(latitude, longitude,

getApplicationContext(), new GeocoderHandler());

} else {

showSettingsAlert();

}

}

});

}

public void showSettingsAlert() {

AlertDialog.Builder alertDialog = new AlertDialog.Builder(

MyActivity.this);

alertDialog.setTitle("SETTINGS");

alertDialog.setMessage("Enable Location Provider! Go to settings menu?");

alertDialog.setPositiveButton("Settings",

new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

Intent intent = new Intent(

Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

MyActivity.this.startActivity(intent);

}

});

alertDialog.setNegativeButton("Cancel",

new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

dialog.cancel();

}

});

alertDialog.show();

}

private class GeocoderHandler extends Handler {

@Override

public void handleMessage(Message message) {

String locationAddress;

switch (message.what) {

case 1:

Bundle bundle = message.getData();

locationAddress = bundle.getString("address");

break;

default:

locationAddress = null;

}

tvAddress.setText(locationAddress);

}

}

}

**AppLocationService.java**

package com.example.exp8;

import android.annotation.SuppressLint;

import android.app.Service;

import android.content.Context;

import android.content.Intent;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.Bundle;

import android.os.IBinder;

public class AppLocationService extends Service implements LocationListener {

protected LocationManager locationManager;

Location location;

private static final long MIN\_DISTANCE\_FOR\_UPDATE = 10;

private static final long MIN\_TIME\_FOR\_UPDATE = 1000 \* 60 \* 2;

public AppLocationService(Context context) {

locationManager = (LocationManager) context

.getSystemService(LOCATION\_SERVICE);

}

@SuppressLint("MissingPermission")

public Location getLocation(String provider) {

if (locationManager.isProviderEnabled(provider)) {

locationManager.requestLocationUpdates(provider,

MIN\_TIME\_FOR\_UPDATE, MIN\_DISTANCE\_FOR\_UPDATE, this);

if (locationManager != null) {

location = locationManager.getLastKnownLocation(provider);

return location;

}

}

return null;

}

@Override

public void onLocationChanged(Location location) {

}

@Override

public void onStatusChanged(String s, int i, Bundle bundle) {

}

@Override

public void onProviderEnabled(String s) {

}

@Override

public void onProviderDisabled(String s) {

}

@Override

public IBinder onBind(Intent intent) {

return null;

}

}

**LocationAddress.java**

package com.example.exp8;

import android.content.Context;

import android.location.Address;

import android.location.Geocoder;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.util.Log;

import java.io.IOException;

import java.util.List;

import java.util.Locale;

public class LocationAddress {

private static final String TAG = "LocationAddress";

public static void getAddressFromLocation(final double latitude, final double longitude,

final Context context, final Handler handler) {

Thread thread = new Thread() {

@Override

public void run() {

Geocoder geocoder = new Geocoder(context, Locale.getDefault());

String result = null;

try {

List<Address> addressList = geocoder.getFromLocation(

latitude, longitude, 1);

if (addressList != null && addressList.size() > 0) {

Address address = addressList.get(0);

StringBuilder sb = new StringBuilder();

for (int i = 0; i < address.getMaxAddressLineIndex(); i++) {

sb.append(address.getAddressLine(i)).append("\n");

}

sb.append(address.getLocality()).append("\n");

sb.append(address.getPostalCode()).append("\n");

sb.append(address.getCountryName());

result = sb.toString();

}

} catch (IOException e) {

Log.e(TAG, "Unable connect to Geocoder", e);

} finally {

Message message = Message.obtain();

message.setTarget(handler);

if (result != null) {

message.what = 1;

Bundle bundle = new Bundle();

result = "Latitude: " + latitude + " \nLongitude: " + longitude +

"\n\nAddress:\n" + result;

bundle.putString("address", result);

message.setData(bundle);

} else {

message.what = 1;

Bundle bundle = new Bundle();

result = "Latitude: " + latitude + " \nLongitude: " + longitude +

"\n Unable to get address for this lat-long.";

bundle.putString("address", result);

message.setData(bundle);

}

message.sendToTarget();

}

}

};

thread.start();

}

}

**activity\_main.xml**

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

<RelativeLayout 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=".MyActivity">

<TextView

android:text="EXPERIMENT 8"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/textView" />

<Button

style="?android:attr/buttonStyleSmall"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Show Location"

android:layout\_marginLeft="20dp"

android:id="@+id/btnGPSShowLocation"

android:layout\_toEndOf="@+id/textView"

android:layout\_marginTop="53dp"

android:layout\_below="@+id/textView"

android:layout\_alignParentStart="true" />

<Button

android:layout\_marginLeft="20dp"

style="?android:attr/buttonStyleSmall"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Show Address"

android:id="@+id/btnShowAddress"

android:layout\_toEndOf="@+id/tvAddress"

android:layout\_below="@+id/btnGPSShowLocation"

android:layout\_alignParentStart="true" />

<TextView

android:layout\_marginLeft="20dp"

android:textSize="20dp"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

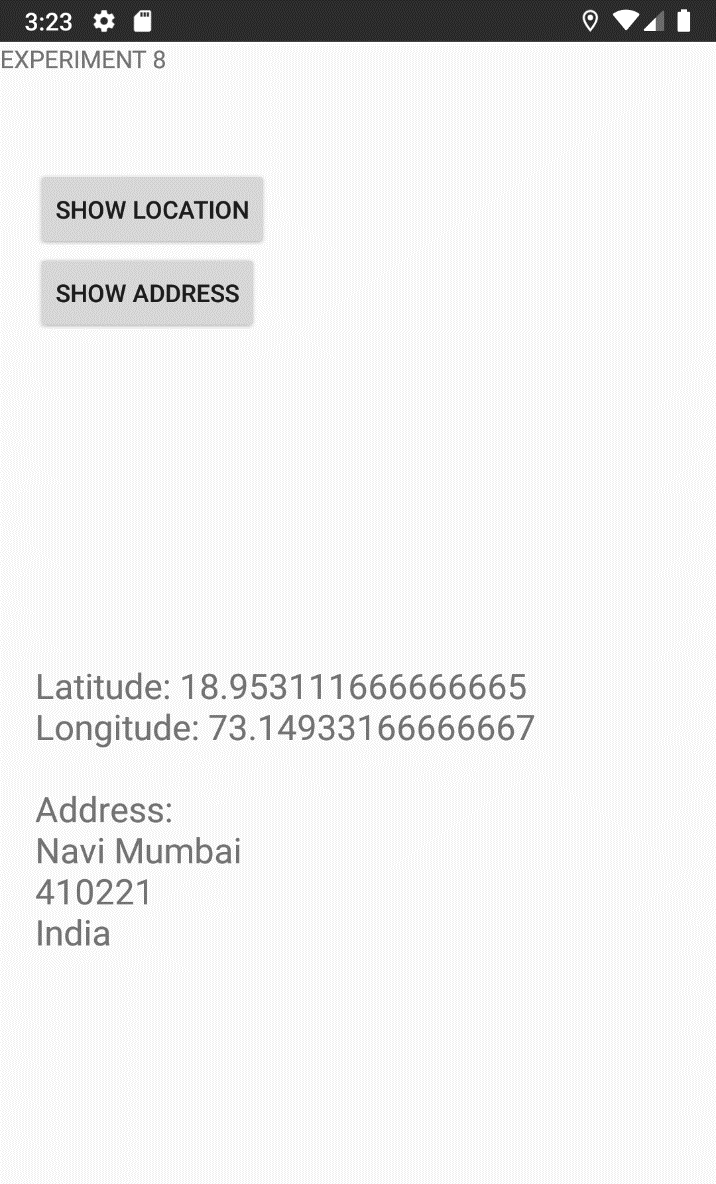
android:id="@+id/tvAddress"

android:layout\_alignParentBottom="true"

android:layout\_marginBottom="134dp"

android:layout\_alignParentEnd="true" />

</RelativeLayout>

**Output:-**