**GpsActivity.java**

**package** android.com.sensor;  
  
  
**import** android.app.Activity;  
**import** android.os.Bundle;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.Toast;  
  
**public class** GpsActivity**extends** Activity {  
GPSHelper**gps**;  
@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
**super**.onCreate(savedInstanceState);  
setContentView(R.layout.***gps\_layout***);  
 Button myLocation = (Button) findViewById(R.id.***btnGPS***);  
myLocation.setOnClickListener(**new** View.OnClickListener() {  
@Override  
**public void** onClick(View view) {  
**gps**= **new** GPSHelper(getApplicationContext());  
**double** latitude = **gps**.getLatitude();  
**double** longitude = **gps**.getLongitude();  
Toast.*makeText*(getApplicationContext(), **"Longitude: "** + longitude + **"\nLatitude: "** + latitude, Toast.***LENGTH\_LONG***).show();  
 }  
 });  
 }  
  
@Override  
**public boolean**onCreateOptionsMenu(Menu menu) {  
*// Inflate the menu; this adds items to the action bar if it is present.*getMenuInflater().inflate(R.menu.***main***, menu);  
**return true**;  
 }  
  
@Override  
**public boolean**onOptionsItemSelected(MenuItem item) {  
*// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.***int**id = item.getItemId();  
**if** (id == R.id.***action\_settings***) {  
**return true**;  
 }  
**return super**.onOptionsItemSelected(item);  
 }  
}

GPSHelper.java

**package** android.com.sensor;  
  
**import** android.location.LocationListener;  
**import** android.location.LocationManager;  
**import** android.app.Service;  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.location.Location;  
**import** android.os.Bundle;  
**import** android.os.IBinder;  
**public class** GPSHelper**extends** Service **implements** LocationListener {  
**private final** Context **context**;  
*// flag for GPS status***booleanisGPSEnabled**= **false**;  
 Location **location**;  
**double latitude**;  
**double longitude**;  
*// minimum distance to update location in meters***private static final long *MIN\_DISTANCE\_CHANGE\_TO\_UPDATES*** = 5; *// 5m  
 // minimum time between updates in milliseconds***private static final long *MIN\_TIME\_ELAPSED\_BETWEEN\_UPDATES*** = 1000 \* 60 \* 1; *// 1 minute  
 // declarea Location Manager***protected** LocationManager**locationManager**;  
**public** GPSHelper(Context context) {  
**this**.**context**= context;  
getLocation();  
 }  
**public** Location getLocation() {  
**try** {  
*// get a reference to the system Location Manager service***locationManager**= (LocationManager) **context**.getSystemService(***LOCATION\_SERVICE***);  
*// get GPS status***isGPSEnabled**= **locationManager**.isProviderEnabled(LocationManager.***GPS\_PROVIDER***);  
*// if GPS is enabled get location via GPS Services***if** (**isGPSEnabled**) {  
**if** (**location** == **null**) {  
**locationManager**.requestLocationUpdates(  
LocationManager.***GPS\_PROVIDER***,  
***MIN\_DISTANCE\_CHANGE\_TO\_UPDATES***,  
***MIN\_TIME\_ELAPSED\_BETWEEN\_UPDATES***, **this**);  
**if** (**locationManager**!= **null**) {  
**location** = **locationManager**.getLastKnownLocation(LocationManager.***GPS\_PROVIDER***);  
**if** (**location** != **null**) {  
**latitude** = **location**.getLatitude();  
**longitude** = **location**.getLongitude();  
 }  
 }  
 }  
 }  
 } **catch** (Exception e) {  
e.printStackTrace();  
 }  
**return location**;  
 }  
**public double** getLatitude(){  
**if**(**location** != **null**){  
**latitude** = **location**.getLatitude();  
 }  
**return latitude**;  
 }  
**public double** getLongitude(){  
**if**(**location** != **null**){  
**longitude** = **location**.getLongitude();  
 }  
**return longitude**;  
 }  
@Override  
**public void** onLocationChanged(Location location) {  
getLocation();  
 }  
  
@Override  
**public void** onProviderDisabled(String provider) {  
 }  
@Override  
**public void** onProviderEnabled(String provider) {  
 }  
@Override  
**public void** onStatusChanged(String provider, **int**status, Bundle extras) {  
 }  
@Override  
**public** IBinderonBind(Intent arg0) {  
**return null**;  
 }  
}

gps\_layout.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayoutxmlns:android="http://schemas.android.com/apk/res/android"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"**>  
  
<**TextViewandroid:id="@+id/text" android:text="Gps"  
style="?android:attr/textAppearanceLarge"  
android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:layout\_centerHorizontal="true"**/>  
  
<**LinearLayoutandroid:id="@+id/l1"  
android:layout\_width="160dp"  
android:layout\_height="80dp"  
android:layout\_centerInParent="true"  
android:background="#d3e4f4"**>  
  
<**TextView  
android:id="@+id/pressTxt"  
android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:gravity="center\_horizontal"  
style="@style/gpsStyle"** />  
  
</**LinearLayout**>  
</**RelativeLayout**>