**package** com.example.krukm.myapplication;  
  
**import** android.graphics.Color;  
**import** android.hardware.Sensor;  
**import** android.hardware.SensorEvent;  
**import** android.hardware.SensorEventListener;  
**import** android.hardware.SensorManager;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
  
**public class** MainActivity **extends** AppCompatActivity **implements** SensorEventListener {  
  
 **private** SensorManager **sensorManager**;  
 **private boolean color** = **false**;  
 **private** TextView **view**;  
 **private long lastUpdate**;  
 **private** SensorManager **mSensorManager**;  
 @Override  
  
 **protected void** onCreate(Bundle savedInstanceState) {  
  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **view** = (TextView)findViewById(R.id.***textView***);  
 **view**.setBackgroundColor(Color.***GREEN***);  
  
 **sensorManager** = (SensorManager) getSystemService(***SENSOR\_SERVICE***);  
 **lastUpdate** = System.*currentTimeMillis*();  
  
 }  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 **if** (event.**sensor**.getType() == Sensor.***TYPE\_ACCELEROMETER***) {  
 getAccelerometer(event);  
 }  
  
 }  
  
 **private void** getAccelerometer(SensorEvent event) {  
 **float**[] values = event.**values**;  
 *// Movement* **float** x = values[0];  
 **float** y = values[1];  
 **float** z = values[2];  
  
 **float** accelationSquareRoot = (x \* x + y \* y + z \* z)  
 / (SensorManager.***GRAVITY\_EARTH*** \* SensorManager.***GRAVITY\_EARTH***);  
 **long** actualTime = event.**timestamp**;  
 **if** (accelationSquareRoot >= 2) *//* {  
 **if** (actualTime - **lastUpdate** < 200) {  
 **return**;  
 }  
 **lastUpdate** = actualTime;  
 Toast.*makeText*(**this**, **"Urządzenie zmieniło położenie"**, Toast.***LENGTH\_SHORT***)  
 .show();  
 **if** (**color**) {  
 **view**.setBackgroundColor(Color.***GREEN***);  
 } **else** {  
 **view**.setBackgroundColor(Color.***RED***);  
 }  
 **color** = !**color**;  
 }  
 }  
  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
  
 }  
  
 @Override  
 **protected void** onResume() {  
 **super**.onResume();  
  
 **sensorManager**.registerListener(**this**,  
 **sensorManager**.getDefaultSensor(Sensor.***TYPE\_ACCELEROMETER***),  
 SensorManager.***SENSOR\_DELAY\_NORMAL***);  
 }  
  
 @Override  
 **protected void** onPause() {  
   
 **super**.onPause();  
 **sensorManager**.unregisterListener(**this**);  
 }  
  
}

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"** >  
  
 <**TextView  
 android:id="@+id/textView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:text="Shake to get a toast and to switch color"** />  
  
</**LinearLayout**>

<**resources**>  
 <**string name="app\_name"**>My Application</**string**>  
 <**string name="hello\_world"**>Hello world!</**string**>  
 <**string name="action\_settings"**>Settings</**string**>  
</**resources**>

**package** com.example.krukm.myapplication;  
  
**import** android.content.Context;  
**import** android.graphics.Canvas;  
**import** android.graphics.Color;  
**import** android.graphics.Paint;  
**import** android.view.View;  
  
**public class** MyCompassView **extends** View {  
  
 **private** Paint **paint**;  
 **private float position** = 0;  
  
 **public** MyCompassView(Context context) {  
 **super**(context);  
 init();  
 }  
  
 **private void** init() {  
 **paint** = **new** Paint();  
 **paint**.setAntiAlias(**true**);  
 **paint**.setStrokeWidth(2);  
 **paint**.setTextSize(25);  
 **paint**.setStyle(Paint.Style.***STROKE***);  
 **paint**.setColor(Color.***WHITE***);  
 }  
  
 @Override  
 **protected void** onDraw(Canvas canvas) {  
 **int** xPoint = getMeasuredWidth() / 2;  
 **int** yPoint = getMeasuredHeight() / 2;  
  
 **float** radius = (**float**) (Math.*max*(xPoint, yPoint) \* 0.6);  
 canvas.drawCircle(xPoint, yPoint, radius, **paint**);  
 canvas.drawRect(0, 0, getMeasuredWidth(), getMeasuredHeight(), **paint**);  
  
   
 canvas.drawLine(xPoint,  
 yPoint,  
 (**float**) (xPoint + radius  
 \* Math.*sin*((**double**) (-**position**) / 180 \* 3.143)),  
 (**float**) (yPoint - radius  
 \* Math.*cos*((**double**) (-**position**) / 180 \* 3.143)), **paint**);  
  
 canvas.drawText(String.*valueOf*(**position**), xPoint, yPoint, **paint**);  
 }  
  
 **public void** updateData(**float** position) {  
 **this**.**position** = position;  
 invalidate();  
 }  
  
}

**package** com.example.krukm.myapplication;  
  
**import** android.content.Context;  
**import** android.hardware.Sensor;  
**import** android.hardware.SensorEvent;  
**import** android.hardware.SensorEventListener;  
**import** android.hardware.SensorManager;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.util.Log;  
**import** android.widget.Toast;  
  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 **private static** SensorManager *sensorService*;  
 **private** MyCompassView **compassView**;  
 **private** Sensor **sensor**;  
  
  
  
  
 @Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 **compassView** = **new** MyCompassView(**this**);  
 setContentView(**compassView**);  
  
 *sensorService* = (SensorManager) getSystemService(Context.***SENSOR\_SERVICE***);  
 **sensor** = *sensorService*.getDefaultSensor(Sensor.***TYPE\_ORIENTATION***);  
 **if** (**sensor** != **null**) {  
 *sensorService*.registerListener(**mySensorEventListener**, **sensor**,  
 SensorManager.***SENSOR\_DELAY\_NORMAL***);  
 Log.*i*(**"Compass MainActivity"**, **"Registerered for ORIENTATION Sensor"**);  
 } **else** {  
 Log.*e*(**"Compass MainActivity"**, **"Registerered for ORIENTATION Sensor"**);  
 Toast.*makeText*(**this**, **"ORIENTATION Sensor not found"**,  
 Toast.***LENGTH\_LONG***).show();  
 finish();  
 }  
 }  
  
 **private** SensorEventListener **mySensorEventListener** = **new** SensorEventListener() {  
  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
 }  
  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 *// angle between the magnetic north direction  
 // 0=North, 90=East, 180=South, 270=West* **float** azimuth = event.**values**[0];  
 **compassView**.updateData(azimuth);  
 }  
 };  
  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 **if** (**sensor** != **null**) {  
 *sensorService*.unregisterListener(**mySensorEventListener**);  
 }  
 }  
  
}