Experiment 22

1.

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity" >
   <TextView
      android:id="@+id/textView"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:text="Shake to switch color" />
</RelativeLayout>
```

MainActivity.java

```
package com.example.exp22_1;
import android.annotation.SuppressLint;
import android.app.Activity;
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.view.View;
import android.widget.Toast;
public class MainActivity extends Activity implements SensorEventListener{
   private SensorManager sensorManager;
   private boolean isColor = false;
   private View view;
   private long lastUpdate;
   @SuppressLint("MissingInflatedId")
   @Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      view = findViewById(R.id.textView);
      view.setBackgroundColor(Color.GREEN);
      sensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
      lastUpdate = System.currentTimeMillis();
   }
   @Override
   public void onAccuracyChanged(Sensor sensor, int accuracy) {}
```

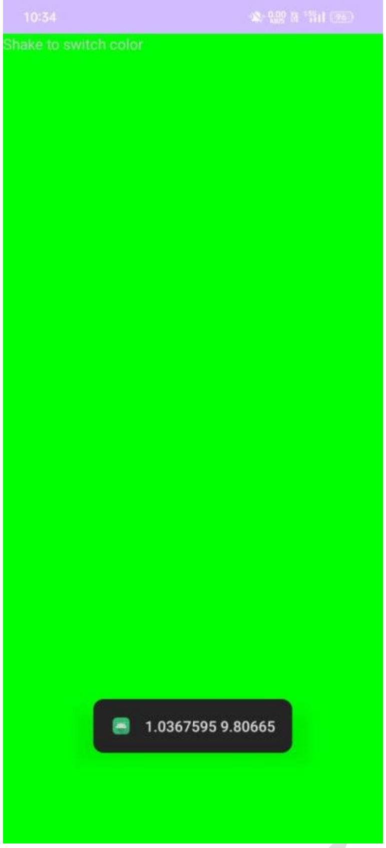
```
  @Override
  public void onSensorChanged(SensorEvent event) {
    if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
      getAccelerometer(event);
    }
  }
  private void getAccelerometer(SensorEvent event) {
    float[] values = event.values;
    float x = values[0];
    float y = values[1];
    float z = values[2];
    float accelationSquareRoot = (x * x + y * y + z * z)
        / (SensorManager.GRAVITY_EARTH * SensorManager.GRAVITY_EARTH);
    long actualTime = System.currentTimeMillis();

    Toast.makeText(getApplicationContext(),String.valueOf(accelationSquareRoot)+" "+
        SensorManager.GRAVITY_EARTH,Toast.LENGTH_SHORT).show();
    if (accelationSquareRoot >= 2)
    {
      if (actualTime - lastUpdate < 200) {
        return;
      }
      lastUpdate = actualTime;
      if (isColor) {
        view.setBackgroundColor(Color.GREEN);
      } else {
        view.setBackgroundColor(Color.RED);
      }
      isColor = !isColor;
    }
  }
  @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);
  }

  private class TYPE_ACC {
  }
```

```
}
```

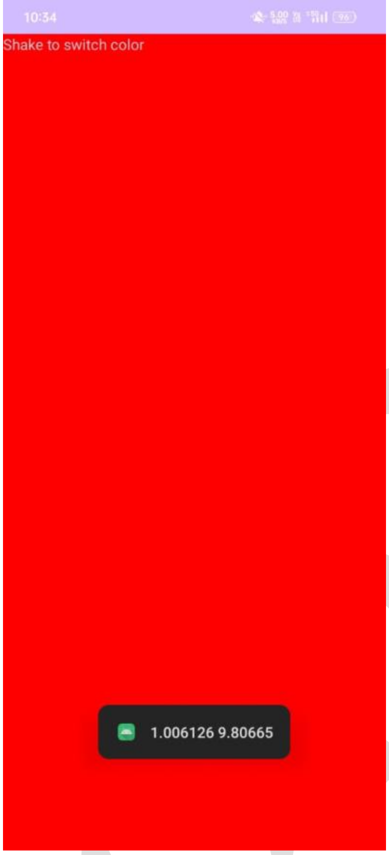
Output:



10:34

Shake to switch color

1.0367595 9.80665



10:34

Shake to switch color

1.006126 9.80665

2.

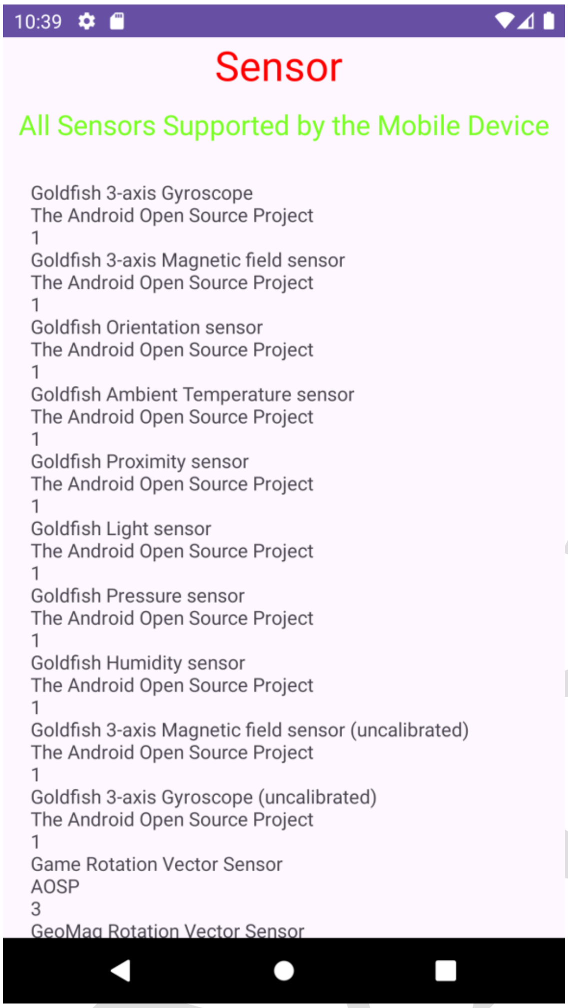
Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:transitionGroup="true">
    <TextView
        android:text="Sensor "
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textview"
        android:textSize="30dp"
        android:textColor="#FF0000"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="All Sensors Supported by the Mobile Device"
        android:layout_marginTop="10dp"
        android:id="@+id/textView1"
        android:layout_below="@+id/textview"
        android:layout_centerHorizontal="true"
        android:textColor="#ff7aff24"
        android:textSize="20sp" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/textView2"
        android:layout_below="@+id/textView1"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentStart="true" />
</RelativeLayout>
```

MainActivity.java

```
package com.example.exp22_2;
import android.app.Activity;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import java.util.List;
import android.hardware.Sensor;
public class MainActivity extends Activity {
    TextView tv1=null;
    private SensorManager mSensorManager;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        tv1 = (TextView) findViewById(R.id.textView2);
        tv1.setVisibility(View.GONE);
        mSensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
        List<Sensor> mList= mSensorManager.getSensorList(Sensor.TYPE_ALL);
        for (int i = 1; i < mList.size(); i++) {
            tv1.setVisibility(View.VISIBLE);
            tv1.append("\n" + mList.get(i).getName() + "\n" +
                    mList.get(i).getVendor() + "\n" + mList.get(i).getVersion());
        }
    }
}
```

Output:



Sensor

All Sensors Supported by the Mobile Device

```
Goldfish 3-axis Gyroscope
The Android Open Source Project
1
Goldfish 3-axis Magnetic field sensor
The Android Open Source Project
1
Goldfish Orientation sensor
The Android Open Source Project
1
Goldfish Ambient Temperature sensor
The Android Open Source Project
1
Goldfish Proximity sensor
The Android Open Source Project
1
Goldfish Light sensor
The Android Open Source Project
1
Goldfish Pressure sensor
The Android Open Source Project
1
Goldfish Humidity sensor
The Android Open Source Project
1
Goldfish 3-axis Magnetic field sensor (uncalibrated)
The Android Open Source Project
1
Goldfish 3-axis Gyroscope (uncalibrated)
The Android Open Source Project
1
Game Rotation Vector Sensor
AOSP
3
GeoMag Rotation Vector Sensor
```