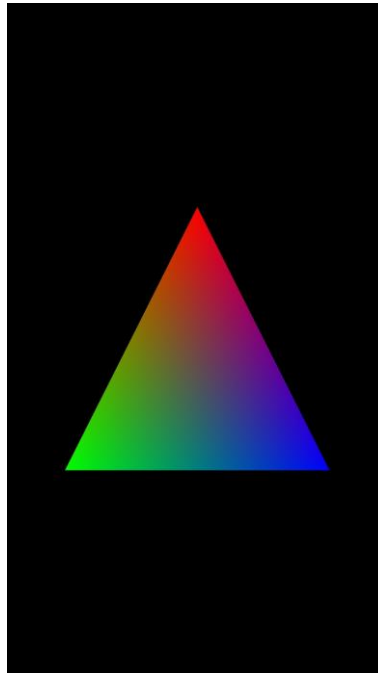




## Android - OpenGL ES 1 - Tutorial 4

### Colored Vertices

With "gl.glColor4f();" we set the color of all the vertices of our model, but if we wanted to set them differently?



Modify the previous project in the following way:

- in the "ObjTriangle" class, add the color variable:

```
private FloatBuffer colorBuffer;
```

add the color information:

```
private float[] colors = {  
    0.0f, 1.0f, 0.0f, 1.0f, // Vertex 0 (Green).  
    0.0f, 0.0f, 1.0f, 1.0f, // Vertex 1 (Blue).  
    1.0f, 0.0f, 0.0f, 1.0f // Vertex 2 (Red).  
};
```

add color conversion to OpenGL format:

```
ByteBuffer vcc = ByteBuffer.allocateDirect(this.colors.length * 4);  
vcc.order(ByteOrder.nativeOrder());  
this.colorBuffer = vcc.asFloatBuffer();  
this.colorBuffer.put(this.colors);  
this.colorBuffer.position(0);
```

create a new method "DrawTriangleRainbow()" for this purpose:

```
public void DrawTriangleRainbow(GL10 gl) {  
    gl.glEnableClientState(GL10.GL_VERTEX_ARRAY);
```



```
gl.glVertexPointer(2, GL10.GL_FLOAT, 0, this.vertexBuffer);
gl.glEnableClientState(GL10.GL_COLOR_ARRAY);
gl.glColorPointer(4, GL10.GL_FLOAT, 0, this.colorBuffer);
gl.glDrawElements(GL10.GL_TRIANGLES, this.indices.length, GL10.GL_UNSIGNED_SHORT,
this.indexBuffer);
gl.glDisableClientState(GL10.GL_VERTEX_ARRAY);
gl.glDisableClientState(GL10.GL_COLOR_ARRAY);
}
```

Compared to the "DrawTriangle()" method, we added:

1. enable the use of colors "gl.glEnableClientState(GL10.GL\_COLOR\_ARRAY);";
2. specific where the colors are "gl.glColorPointer(4, GL10.GL\_FLOAT, 0, this.colorBuffer);";
3. disable the use of colors "gl.glDisableClientState(GL10.GL\_COLOR\_ARRAY);".

- in the "onDrawFrame()" method we use:

```
gl.glLoadIdentity();
gl.glTranslatef(this.width / 2.0f, this.height / 2.0f, 0.0f);
gl.glScalef(500.0f, 500.0f, 0.0f);
this.objtriangle.DrawTriangleRainbow(gl);
```