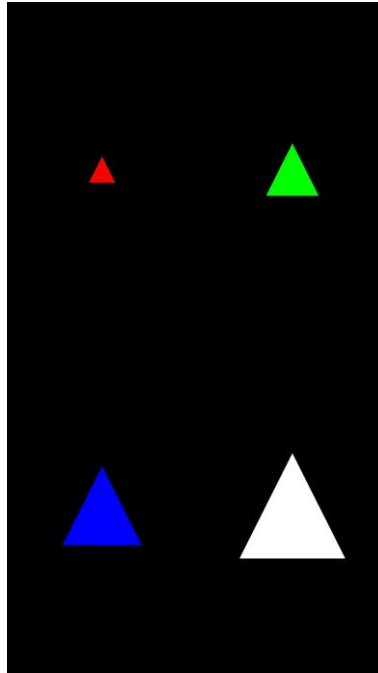


Android - OpenGL ES 1 - Tutorial 3

Draw Triangles

The purpose of this tutorial is to draw 4 triangles as in the following screenshot:



Modify the previous project in the following way:

- in the "ObjTriangle" class, the "DrawTriangle()" method becomes:

```
public void DrawTriangle(GL10 gl, int color) {
    gl.glEnableClientState(GL10.GL_VERTEX_ARRAY);
    gl.glVertexPointer(2, GL10.GL_FLOAT, 0, this.vertexBuffer);
    if (color == 1) // Red.
        gl.glColor4f(1.0f, 0.0f, 0.0f, 1.0f);
    else if (color == 2) // Green.
        gl.glColor4f(0.0f, 1.0f, 0.0f, 1.0f);
    else if (color == 3) // Blue.
        gl.glColor4f(0.0f, 0.0f, 1.0f, 1.0f);
    else // White.
        gl.glColor4f(1.0f, 1.0f, 1.0f, 1.0f);
    gl.glDrawElements(GL10.GL_TRIANGLES, this.indices.length, GL10.GL_UNSIGNED_SHORT,
this.indexBuffer);
    gl.glDisableClientState(GL10.GL_VERTEX_ARRAY);
}
```

Here we have added the "color" variable that we pass during the call to specify the color to be used.

- the "onDrawFrame()" method becomes:

```
@Override
public void onDrawFrame(GL10 gl) {
    gl.glClear(GL10.GL_COLOR_BUFFER_BIT);

    gl.glLoadIdentity();
    gl.glTranslatef(this.width / 4.0f, this.height / 4.0f * 3.0f, 0.0f);
```



```
gl.glScalef(50.0f, 50.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 1);

gl.glLoadIdentity();
gl.glTranslatef(this.width / 4.0f * 3.0f, this.height / 4.0f * 3.0f, 0.0f);
gl.glScalef(100.0f, 100.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 2);

gl.glLoadIdentity();
gl.glTranslatef(this.width / 4.0f, this.height / 4.0f, 0.0f);
gl.glScalef(150.0f, 150.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 3);

gl.glLoadIdentity();
gl.glTranslatef(this.width / 4.0f * 3.0f, this.height / 4.0f, 0.0f);
gl.glScalef(200.0f, 200.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 0);
}
```

Note the reset of the modelview matrix with "gl.glLoadIdentity();" before each drawing. If we do not reset the matrix, translate or scale will be performed on the basis of the previous operation.

The code:

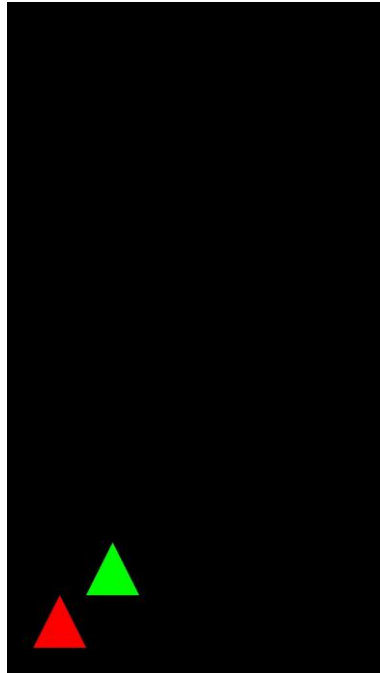
```
gl.glLoadIdentity();
gl.glTranslatef(100.0f, 100.0f, 0.0f);
gl.glScalef(100.0f, 100.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 1);

gl.glLoadIdentity();
gl.glTranslatef(200.0f, 200.0f, 0.0f);
gl.glScalef(100.0f, 100.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 2);
```

and the code:

```
gl.glLoadIdentity();
gl.glScalef(100.0f, 100.0f, 0.0f);
gl.glTranslatef(1.0f, 1.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 1);
gl.glTranslatef(1.0f, 1.0f, 0.0f);
this.objtriangle.DrawTriangle(gl, 2);
```

produce the same result:



In the first code, for each triangle, we reset, translate, scale and draw.

In the second code, we reset, scale (obtaining a basic unit of "100"), translate by 1 (that is 100), draw, translate again by 1 (that is an other 100) on the previous translate, draw.