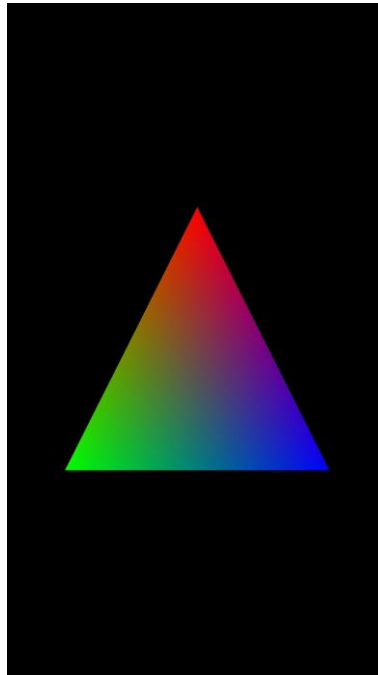


Android - OpenGL ES 1 - Tutorial 4

Vertici Colorati

Con "gl.glColor4f();" impostiamo il colore di tutti i vertici del nostro modello, ma se volessimo impostarli in modo differente?



Modifichiamo il progetto precedente nel seguente modo:

- nella classe "ObjTriangle", aggiungiamo la variabile per il colore:

```
private FloatBuffer colorBuffer;
```

aggiungiamo le informazioni dei colori:

```
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).  
};
```

aggiungiamo la conversione dei colori in formato utile ad OpenGL:

```
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);
```

creiamo un nuovo metodo "DrawTriangleRainbow()" per tale scopo:

```
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);
}
```

Rispetto al metodo "DrawTriangle()", abbiamo aggiunto:

1. abilito l'uso dei colori "gl.glEnableClientState(GL10.GL_COLOR_ARRAY);";
2. specifico dove sono i colori "gl.glColorPointer(4, GL10.GL_FLOAT, 0, this.colorBuffer);";
3. disabilito l'uso dei colori "gl.glDisableClientState(GL10.GL_COLOR_ARRAY);".

- nel metodo "onDrawFrame()" usiamo:

```
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);
```