

Render

```
drawTriangle(points, std::string texture);  
drawRectangle(points, texture);  
drawCircle(position, radius, texture);  
drawLine(x, y, texture);
```

```
drawTriangle(points, color);  
drawRectangle(points, color);  
drawCircle(position, radius, color);  
drawLine(x, y, color);
```

```
drawMultiColorTriangle(points, color1,  
color2, color3);  
drawMultiColorRectangle(points, color1,  
color2, color3, color4);  
drawMultiColorCircle(points, color1,  
color2);  
drawMultiColorLinepoints, color1,  
color2);
```

```
render();  
initRender(GLFWwindow* w);  
uninitRender();  
clear();
```

Color

```
enum colors;  
Color(R,G,B,A = 1.0f);  
public:  
float R, G,B, A;
```

TextureManager

```
unordered_map<std::string, GLuint>  
textures;  
getTexture(std::string name);  
private:  
loadTexture(std::string name, std::string  
filePath);  
deleteTexture(std::string fileName);  
vector<*Sprite> sprites;
```

Sprite

```
Sprite( );
```

```
GLuint textureID[];  
int N_images;
```