RT DESIGN 101

OBJECTS OF INTERESTS

* Lights
* Shadows
* Texture
* Reflection
* Shapes
* Scene
* Camera
* Background

OBJECTS OF INTERESTS SORTED BY PRIORITY

* Scene
* Shapes
* Camera
* Lights
* Textures
* Reflection
* Shadows
* Background

Vector3 (namespace: utility)

X,y,z

*Float Magnitude()*

*Vector3 Normalize()*

SCENE (namespace: SceneObjects)

* Height and Width (floats)
* Background (Vector3)

SHAPE (namespace: Shapes)

TRIANGLE

VertexA (Vector3),

VertexB (Vector3),

VertexC (Vector3)

RECTANGLE : TRIANGLE

VertexD (Vector3)

SPHERE

Center (Vector3)

Radius (float)

CAMERAS (namespace: Camera)

CAMERA

* + FocalLength (float)
  + Position (Vector3)
  + Target (Vector3)
  + UpDirection (Vector3)
  + FarPlane (float)
  + FieldOfView (float) – [Radians]
  + C1, C2, C3, C4 (Vector3), der
  + PixelWidth. PixelHeight (float) der
  + View Direction (vector3) der
  + Camera Up (Vector3) der
  + Side Direction (Vector3) der
  + Frame Width, Height (Vector3) der

PERSPECTIVE\_CAMERA : CAMERA

ORTHOGRAPHIC\_CAMERA : CAMERA

MATERIALS (namespace: Materials)  
 MATERIAL

Id (int)

* Specularities (List<float>)
* Type (enum) [SOLID, GRADIENT, CHECKERED,STRIPE]

Abstract Vector3 getAmbientColor(Shape,Vector3)

Abstract Vector3 getDiffuseColor(Shape,Vector3)

Abstract Vector3 getSpecularColor(Shape,Vector3)

Abstract float getSpecularity(Shape,Vector3)

COLOR\_MATERIAL : MATERIAL

* AmbientColors (List<Vector3>)
* DiffuseColors (List<Vector3>)
* SpecularColors (List<Vector3>)

LIGHTS (namespace: LIGHTS)

LIGHT

Id (int)

* Direction
* Color

Vector3 getDirection(Vector3 coordinate)

DIRECTIONAL\_LIGHT : LIGHT

POINT\_LIGHT : LIGHT

* Origin

SPOT\_LIGHT : POINT\_LIGHT

* InnerRadius
* FallOfRadius
* FallOf