TriangleMesh3

```
  <<friend>> TensorProductSurface3: class
# _usage_flag: GLenum
# _vbo_positions: GLuint
# _vbo_normals: GLuint
# _vbo_colors: GLuint
# _vbo_tex_coordinates: GLuint
# _vbo_indices: GLuint
# _leftmost_vertex: Cartesian3
# _rightmost_vertex: Cartesian3
# _position: std::vector<Cartesian3>
# _normal: std::vector<Cartesian3>
# _color: std::vector<Color4>
# _tex: std::vector<TCoordinate4>
# _face: std::vector<TriangularFace>
```

```
+ TriangleMesh3(vertex_count: GLint = 0, face_count: GLint = 0, usage_flag: GLenum = GL_STATIC_DRAW)
+ TriangleMesh3(mesh: const TriangleMesh3&)
+ operator =(rhs: const TriangleMesh3&): TriangleMesh3&
+ deleteVertexBufferObjects(): GLvoid
+ <<const>> render(program: const ShaderProgram&, render_mode: GLenum = GL_TRIANGLES): GLboolean
+ <<const>> render(render_mode: GLenum = GL_TRIANGLES,
                   vec3_position_location: GLint = 0,
                   vec3_normal_location: GLint = 1,
                   vec4_color_location: GLint = 2,
                   vec4_texture_location: GLint = 3): GLboolean
+ updateVertexBufferObjects(usage_flag: GLenum = GL_STATIC_DRAW)
+ loadFromOFF(file_name: const std::string&,
              translate_and_scale_to_unit_cube: GLboolean = GL_FALSE): GLboolean
+ <<const>> mapPositionBuffer(access_flag: GLenum = GL_READ_ONLY): GLfloat*
+ <<const>> mapNormalBuffer(access_flag: GLenum = GL_READ_ONLY): GLfloat*
+ <<const>> mapColorBuffer(access_flag: GLenum = GL_READ_ONLY): GLfloat*
+ <<const>> mapTextureBuffer(access_flag: GLenum = GL_READ_ONLY): GLfloat*
+ <<const>> unmapPositionBuffer(): GLvoid
+ <<const>> unmapNormalBuffer(): GLvoid
+ <<const>> unmapTextureBuffer(): GLvoid
+ <<const>> vertexCount(): GLint
+ <<const>> faceCount(): GLint
+ <<const>> position(index: GLint): const Cartesian3&
+ <<const>> normal(index: GLint): const Cartesian3&
+ <<const>> color(index: GLint): const Color4&
+ <<const>> texture(index: GLint): const TCoordinate4&
+ <<const>> face(index: GLint): const TriangularFace&
+ position(index: GLint): Cartesian3&
+ normal(index: GLint): Cartesian3&
+ color(index: GLint): Color4&
+ texture(index: GLint): TCoordinate4&
+ face(index: GLint): TriangularFace&
+ <<const>> clone(): TriangleMesh3*
+ ~TriangleMesh3()
  <<friend>> operator <<(lhs: std::ostream&, rhs: const TriangleMesh3&): std::ostream&
  <<friend>> operator >>(lhs: std::istream&, rhs: TriangleMesh3&): std::istream&
```