**海南大学计算机科学与技术学院**

**《计算机图形学》课内实验报告五**

****

**班 级：计算机科学与技术2021-3班**

**成 员： 李季鸿 20213002624**

**报告名称： 实验报告五**

**指导老师： 高新瑞**

**完成日期： 2023年10月19日**

**《计算机图形学》实验报告五**

**学生姓名：李季鸿 班级：21级计科3班 学号：20213002624**

**实验地点：9-202 指导教师：高新瑞**

**实验日期：**2023.10.19  **实验课时：2学时**

**实验环境：**Windows 10+JDK1.8+记事本+IntelliJ IDEA

**一、实验目的**

**实验一目的：**

这个示例的学习目的是介绍如何使用Java 3D库创建一个简单的三维图形，并学习以下内容：

如何创建Java 3D应用程序。

如何使用IndexedTriangleArray类来定义三维模型的几何和颜色属性。

如何设置背景和光照效果。

如何使用鼠标交互控制旋转、缩放和平移。

**实验二目的：**

如何创建Java 3D应用程序。

如何使用QuadArray类来定义四边面的几何和颜色属性。

如何设置背景和光照效果。

如何使用鼠标交互控制旋转、缩放和平移。

**实验三目的：**

如何创建Java 3D应用程序。

如何添加透明物体并设置透明属性。

如何添加环境光源，改善渲染效果。

如何使用鼠标交互控制旋转、缩放和平移。

**二、实验过程**

## 实验内容一：

（1）代码

package week8\_fifth.\_3\_13;  
  
import com.sun.j3d.utils.applet.MainFrame;  
import com.sun.j3d.utils.behaviors.mouse.MouseRotate;  
import com.sun.j3d.utils.behaviors.mouse.MouseTranslate;  
import com.sun.j3d.utils.behaviors.mouse.MouseZoom;  
import com.sun.j3d.utils.universe.SimpleUniverse;  
  
import javax.media.j3d.\*;  
import javax.vecmath.Color3f;  
import javax.vecmath.Point3d;  
import javax.vecmath.Vector3f;  
import java.applet.Applet;  
import java.awt.\*;  
  
/\*\*  
 \* \\* Created with IntelliJ IDEA.  
 \* \\* @ProjectName: Computer graphics  
 \* \\* @FileName: IndexedTriangleArray\_1  
 \* \\* @author: li-jihong  
 \* \\* Date: 2023-11-02 12:56  
 \*/  
  
public class IndexedTriangleArray\_1 extends Applet {  
 public IndexedTriangleArray\_1() {  
 setLayout(new BorderLayout());  
 GraphicsConfiguration gc = SimpleUniverse.getPreferredConfiguration();  
 Canvas3D c = new Canvas3D(gc);  
 add("Center", c);  
 BranchGroup BranchGroupScene = createBranchGroup();  
 SimpleUniverse u = new SimpleUniverse(c);  
 u.getViewingPlatform().setNominalViewingTransform();  
 u.addBranchGraph(BranchGroupScene);  
 }  
  
 public static void main(String[] args) {  
 new MainFrame(new IndexedTriangleArray\_1(), 450, 450);  
 }  
  
 public BranchGroup createBranchGroup() {  
 BranchGroup BranchGroupRoot = new BranchGroup();  
 BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);  
 Color3f bgColor = new Color3f(1.0f, 1.0f, 1.0f);  
 Background bg = new Background(bgColor);  
 bg.setApplicationBounds(bounds);  
 BranchGroupRoot.addChild(bg);  
 Color3f directionalColor = new Color3f(1.f, 0.f, 0.f);  
 Vector3f vec = new Vector3f(0.f, 0.f, -1.0f);  
 DirectionalLight directionalLight = new DirectionalLight(directionalColor, vec);  
 directionalLight.setInfluencingBounds(bounds);  
 BranchGroupRoot.addChild(directionalLight);  
 TransformGroup transformgroup = new TransformGroup();  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_WRITE);  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_READ);  
 BranchGroupRoot.addChild(transformgroup);  
 MouseRotate mouserotate = new MouseRotate();  
 mouserotate.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mouserotate);  
 mouserotate.setSchedulingBounds(bounds);  
 MouseZoom mousezoom = new MouseZoom();  
 mousezoom.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mousezoom);  
 mousezoom.setSchedulingBounds(bounds);  
 MouseTranslate mousetranslate = new MouseTranslate();  
 mousetranslate.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mousetranslate);  
 mousetranslate.setSchedulingBounds(bounds);  
 transformgroup.addChild(new ShapeIndexedTriangleArray());  
 BranchGroupRoot.compile();  
 return BranchGroupRoot;  
 }  
}  
  
class ShapeIndexedTriangleArray extends Shape3D {  
 public ShapeIndexedTriangleArray() {  
 int vertexesCount = 12;  
 int indexcount = 6;  
 int[] index = {0, 1, 3, 5, 7, 9};  
 float vertexes[] = {-.9f, 0.8f, 0.0f, -0.8f, -0.8f, 0.2f,  
 -0.6f, -0.4f, -0.2f, -0.4f, -0.9f, 0.2f,  
 -0.2f, 0.8f, -0.2f, 0.f, -0.8f, 0.2f,  
 0.2f, -0.5f, 0.0f, 0.4f, 0.6f, -0.5f,  
 0.6f, -0.8f, -0.3f, 0.8f, -0.9f, -0.2f,  
 0.9f, 0.7f, -0.2f, 1.1f, -0.8f, -0.3f};  
 float colors[] = {0.0f, 0.5f, 1.0f, 0.0f, 0.5f, 1.0f,  
 0.0f, 0.8f, .0f, 1.0f, 0.0f, 0.3f,  
 0.0f, 1.0f, 0.5f, 0.9f, 1.0f, 0.0f,  
 0.5f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,  
 1.0f, 0.5f, 0.0f, 1.0f, 0.0f, 0.5f,  
 1.0f, 0.8f, 0.0f, 1.0f, 0.5f, 0.0f};  
 IndexedTriangleArray indextrianglearray = new IndexedTriangleArray(vertexesCount,  
 IndexedTriangleArray.COORDINATES | IndexedTriangleArray.COLOR\_3, indexcount);  
 indextrianglearray.setCoordinates(0, vertexes);  
 indextrianglearray.setColors(0, colors);  
 indextrianglearray.setCoordinateIndices(0, index);  
 indextrianglearray.setColorIndices(0, index);  
 PolygonAttributes polygonattributes = new PolygonAttributes();  
 polygonattributes.setCullFace(PolygonAttributes.CULL\_NONE);  
 Appearance app = new Appearance();  
 app.setPolygonAttributes(polygonattributes);  
 this.setGeometry(indextrianglearray);  
 this.setAppearance(app);  
 }  
}

（2）结果截图

|  |
| --- |
|  |
| IndexedTriangleArray |

## 实验内容二：

（1）代码

package week8\_fifth;  
  
import com.sun.j3d.utils.applet.MainFrame;  
import com.sun.j3d.utils.behaviors.mouse.MouseRotate;  
import com.sun.j3d.utils.behaviors.mouse.MouseTranslate;  
import com.sun.j3d.utils.behaviors.mouse.MouseZoom;  
import com.sun.j3d.utils.universe.SimpleUniverse;  
  
import javax.media.j3d.\*;  
import javax.vecmath.Color3f;  
import javax.vecmath.Point3d;  
import javax.vecmath.Vector3f;  
import java.applet.Applet;  
import java.awt.\*;  
  
/\*\*  
 \* \\* Created with IntelliJ IDEA.  
 \* \\* @ProjectName: Computer graphics  
 \* \\* @FileName: QuadArray\_1  
 \* \\* @author: li-jihong  
 \* \\* Date: 2023-11-02 13:08  
 \*/  
  
public class QuadArray\_1 extends Applet {  
 public QuadArray\_1() {  
 setLayout(new BorderLayout());  
 GraphicsConfiguration gc = SimpleUniverse.getPreferredConfiguration();  
 Canvas3D c = new Canvas3D(gc);  
 add("Center", c);  
 BranchGroup BranchGroupScene = createBranchGroup();  
 SimpleUniverse u = new SimpleUniverse(c);  
 u.getViewingPlatform().setNominalViewingTransform();  
 u.addBranchGraph(BranchGroupScene);  
 }  
  
 public static void main(String[] args) {  
 new MainFrame(new QuadArray\_1(), 450, 450);  
 }  
  
 public BranchGroup createBranchGroup() {  
 BranchGroup BranchGroupRoot = new BranchGroup();  
 BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);  
 Color3f bgColor = new Color3f(1.0f, 1.0f, 1.0f);  
 Background bg = new Background(bgColor);  
 bg.setApplicationBounds(bounds);  
 BranchGroupRoot.addChild(bg);  
 Color3f directionalColor = new Color3f(1.f, 0.f, 0.f);  
 Vector3f vec = new Vector3f(0.f, 0.f, -1.0f);  
 DirectionalLight directionalLight = new DirectionalLight(directionalColor, vec);  
 directionalLight.setInfluencingBounds(bounds);  
 BranchGroupRoot.addChild(directionalLight);  
 TransformGroup transformgroup = new TransformGroup();  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_WRITE);  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_READ);  
 BranchGroupRoot.addChild(transformgroup);  
 MouseRotate mouserotate = new MouseRotate();  
 mouserotate.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mouserotate);  
 mouserotate.setSchedulingBounds(bounds);  
 MouseZoom mousezoom = new MouseZoom();  
 mousezoom.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mousezoom);  
 mousezoom.setSchedulingBounds(bounds);  
 MouseTranslate mousetranslate = new MouseTranslate();  
 mousetranslate.setTransformGroup(transformgroup);  
 BranchGroupRoot.addChild(mousetranslate);  
 mousetranslate.setSchedulingBounds(bounds);  
 transformgroup.addChild(new ShapeQuadArray());  
 BranchGroupRoot.compile();  
 return BranchGroupRoot;  
 }  
}  
  
class ShapeQuadArray extends Shape3D {  
 public ShapeQuadArray() {  
 int vertexCount = 12;  
 float vertexes[] = {-0.8f, 0.9f, 0.0f, -0.8f, -0.8f, 0.f,  
 -0.6f, -0.8f, 0.f, -0.6f, 0.9f, 0.f,  
 -0.4f, 0.9f, 0.f, -0.4f, -0.7f, -0.9f,  
 0.4f, -0.8f, 0.f, 0.4f, 0.8f, 0.0f,  
 0.5f, 0.8f, 0.f, 0.6f, -0.8f, 0.0f,  
 0.8f, -0.7f, 0.f, 0.8f, 0.8f, 0.f};  
 float colors[] = {0.0f, 0.5f, 1.0f, 0.0f, 0.5f, 1.0f,  
 0.0f, 0.8f, .0f, 1.0f, 0.0f, 0.3f,  
 0.0f, 1.0f, 0.5f, 0.9f, 1.0f, 0.0f,  
 0.5f, 0.0f, 1.0f, 0.0f, 0.5f, 1.0f,  
 1.0f, 0.5f, 0.0f, 1.0f, 0.0f, 0.5f,  
 1.0f, 0.8f, 0.0f, 1.0f, 0.5f, 0.0f};  
 QuadArray quadarray = new QuadArray(vertexCount,  
 QuadArray.COORDINATES | QuadArray.COLOR\_3);  
 quadarray.setCoordinates(0, vertexes);  
 quadarray.setColors(0, colors);  
 PolygonAttributes polygonattributes = new PolygonAttributes();  
 polygonattributes.setCullFace(PolygonAttributes.CULL\_NONE);  
 //polygonattributes.setCullFace(PolygonAttributes.CULL\_BACK);  
 //polygonattributes.setCullFace(PolygonAttributes.CULL\_FRONT);  
 Appearance app = new Appearance();  
 app.setPolygonAttributes(polygonattributes);  
 this.setGeometry(quadarray);  
 this.setAppearance(app);  
 }  
}

（2）结果截图

|  |  |
| --- | --- |
|  |  |

## 实验内容三：

（1）代码

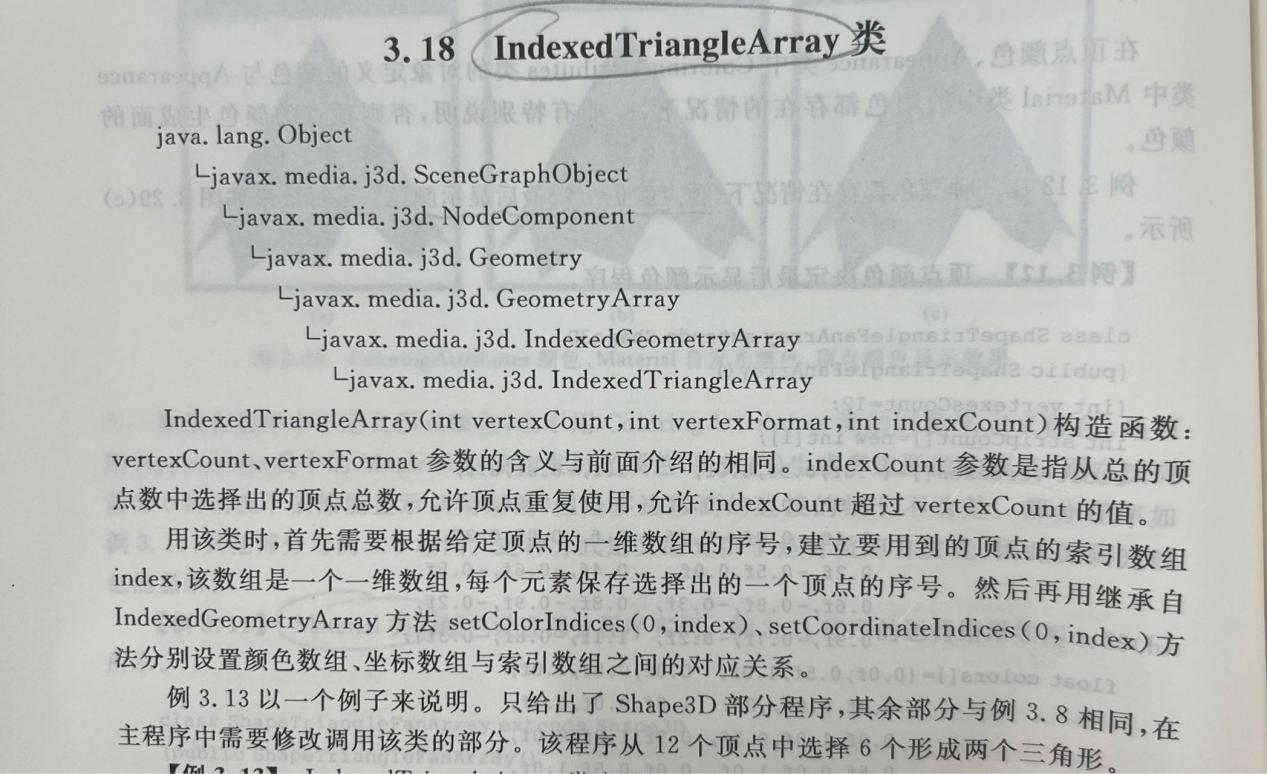
package week8\_fifth.\_3\_16;  
  
/\*\*  
 \* \\* Created with IntelliJ IDEA.  
 \* \\* @ProjectName: Computer graphics  
 \* \\* @FileName: TransparencyAttributess  
 \* \\* @author: li-jihong  
 \* \\* Date: 2023-11-02 13:13  
 \*/  
  
import com.sun.j3d.utils.applet.MainFrame;  
import com.sun.j3d.utils.behaviors.mouse.MouseRotate;  
import com.sun.j3d.utils.behaviors.mouse.MouseTranslate;  
import com.sun.j3d.utils.behaviors.mouse.MouseZoom;  
import com.sun.j3d.utils.geometry.Box;  
import com.sun.j3d.utils.geometry.Sphere;  
import com.sun.j3d.utils.universe.SimpleUniverse;  
  
import javax.media.j3d.\*;  
import javax.vecmath.Color3f;  
import javax.vecmath.Point3d;  
import javax.vecmath.Vector3f;  
import java.applet.Applet;  
import java.awt.\*;  
  
public class TransparencyAttributess extends Applet {  
 public TransparencyAttributess() {  
 setLayout(new BorderLayout());  
 GraphicsConfiguration gc = SimpleUniverse.getPreferredConfiguration();  
 Canvas3D c = new Canvas3D(gc);  
 add("Center", c);  
 BranchGroup BranchGroupScene = createBranchGroupSceneGraph();  
 SimpleUniverse u = new SimpleUniverse(c);  
 u.getViewingPlatform().setNominalViewingTransform();  
 u.addBranchGraph(BranchGroupScene);  
 }  
  
 public static void main(String[] args) {  
 new MainFrame(new TransparencyAttributess(), 300, 300);  
 }  
  
 public BranchGroup createBranchGroupSceneGraph() {  
 BranchGroup BrachGroupRoot = new BranchGroup();  
 BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);  
 Color3f bgColor = new Color3f(1.0f, 1.0f, 1.0f);  
 Background bg = new Background(bgColor);  
 bg.setApplicationBounds(bounds);  
 BrachGroupRoot.addChild(bg);  
 Color3f directionalColor = new Color3f(1.f, 1.f, 1.f);  
 Vector3f vec = new Vector3f(-1.f, -1.f, -1.0f);  
 DirectionalLight directionalLight = new DirectionalLight(directionalColor, vec);  
 directionalLight.setInfluencingBounds(bounds);  
 BrachGroupRoot.addChild(directionalLight);  
 Appearance app1 = new Appearance();  
 Material material1 = new Material();  
 material1.setDiffuseColor(new Color3f(1.0f, .0f, 0.0f));  
 app1.setMaterial(material1);  
//定义球体的透明度，透明模式选NICEST=1，透明度0.6f  
 TransparencyAttributes transparence = new TransparencyAttributes(1, .6f);  
 app1.setTransparencyAttributes(transparence);  
 Appearance app2 = new Appearance();  
 Material material2 = new Material();  
 material2.setDiffuseColor(new Color3f(.0f, 1.0f, 0.0f));  
 app2.setMaterial(material2);  
 TransparencyAttributes transparence1 = new TransparencyAttributes(1, .8f);  
 app2.setTransparencyAttributes(transparence1);  
 Appearance app3 = new Appearance();  
 Material material3 = new Material();  
 material3.setDiffuseColor(new Color3f(.0f, .0f, 1.0f));  
 app3.setMaterial(material3);  
 app3.setTransparencyAttributes(transparence1);  
 Appearance app4 = new Appearance();  
 Material material4 = new Material();  
 material4.setDiffuseColor(new Color3f(.0f, 1.0f, 1.0f));  
 app4.setMaterial(material2);  
 app4.setTransparencyAttributes(transparence1);  
 TransformGroup transformgroup = new TransformGroup();  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_WRITE);  
 transformgroup.setCapability(TransformGroup.ALLOW\_TRANSFORM\_READ);  
 BrachGroupRoot.addChild(transformgroup);  
 MouseRotate mouserotate = new MouseRotate();  
 mouserotate.setTransformGroup(transformgroup);  
 BrachGroupRoot.addChild(mouserotate);  
 mouserotate.setSchedulingBounds(bounds);  
 MouseZoom mousezoom = new MouseZoom();  
 mousezoom.setTransformGroup(transformgroup);  
 BrachGroupRoot.addChild(mousezoom);  
 mousezoom.setSchedulingBounds(bounds);  
 MouseTranslate mousetranslate = new MouseTranslate();  
 mousetranslate.setTransformGroup(transformgroup);  
 BrachGroupRoot.addChild(mousetranslate);  
 mousetranslate.setSchedulingBounds(bounds);  
 TransformGroup tg1 = new TransformGroup();  
 tg1.addChild(new Sphere(.9f, 1, 100, app1));  
 Transform3D t = new Transform3D();  
 t.setTranslation(new Vector3f(-0.2f, 0.1f, 0.2f));  
 TransformGroup tg2 = new TransformGroup(t);  
 tg2.addChild(new Box(0.2f, 0.2f, 0.2f, app2));  
 Transform3D t1 = new Transform3D();  
 t1.setTranslation(new Vector3f(-0.2f, 0.1f, 0.2f));  
 TransformGroup tg3 = new TransformGroup(t1);  
 tg3.addChild(new Sphere(.6f, 1, 100, app3));  
 Transform3D t2 = new Transform3D();  
 t2.setTranslation(new Vector3f(0.4f, 0.2f, -0.4f));  
 TransformGroup tg4 = new TransformGroup(t2);  
 tg4.addChild(new Sphere(.3f, 1, 100, app4));  
 transformgroup.addChild(tg1);  
 transformgroup.addChild(tg2);  
 transformgroup.addChild(tg3);  
 transformgroup.addChild(tg4);  
 BrachGroupRoot.compile();  
 return BrachGroupRoot;  
 }  
}

（2）结果截图

|  |  |
| --- | --- |
|  |  |

**三、实验总结**

**实验一总结：**

****

创建Java 3D应用程序：可以学习如何创建一个Java 3D应用程序，包括设置窗口和创建3D场景。

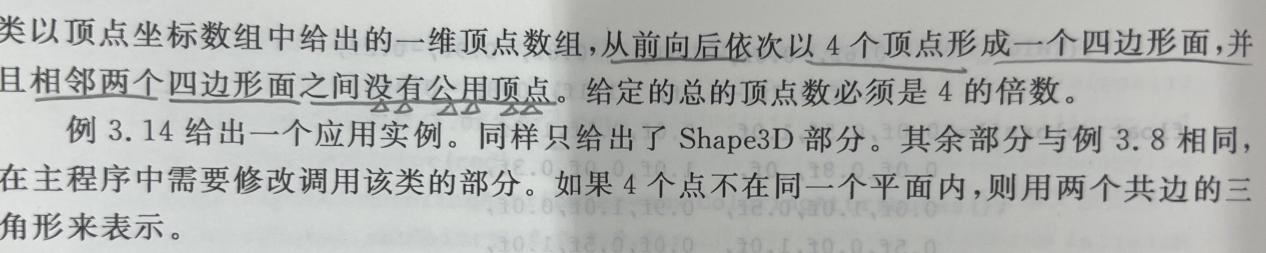
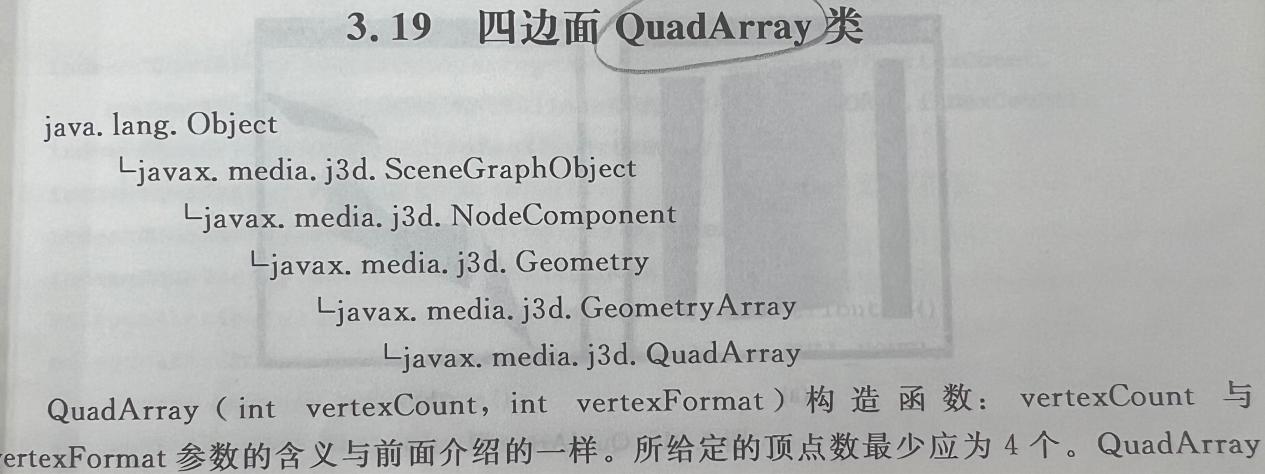
使用IndexedTriangleArray类：了解了如何使用IndexedTriangleArray类来定义三维模型的顶点坐标和颜色信息。这是构建3D模型的基本步骤之一。

背景和光照：学会了如何设置背景颜色以及如何添加定向光源来影响模型的照明效果。这是创建逼真三维场景的关键。

鼠标交互：通过添加鼠标旋转、缩放和平移的交互功能，可以学习如何使用户能够以交互方式浏览和操作三维场景。

渲染和显示：最后，学会了如何将创建的3D模型添加到场景中，并通过设置外观（Appearance）来定义模型的渲染属性。

**实验二总结：**

****

创建Java 3D应用程序：可以学习如何创建一个Java 3D应用程序，包括设置窗口和创建3D场景。

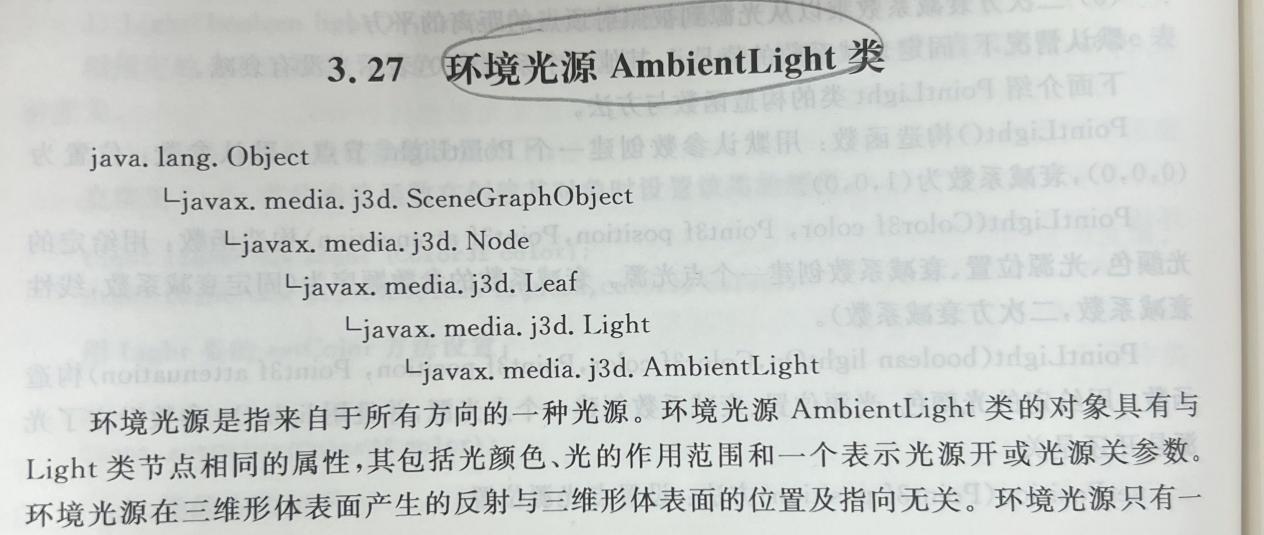
使用QuadArray类：了解了如何使用QuadArray类来定义四边面的顶点坐标和颜色信息。这是构建3D模型的基本步骤之一。

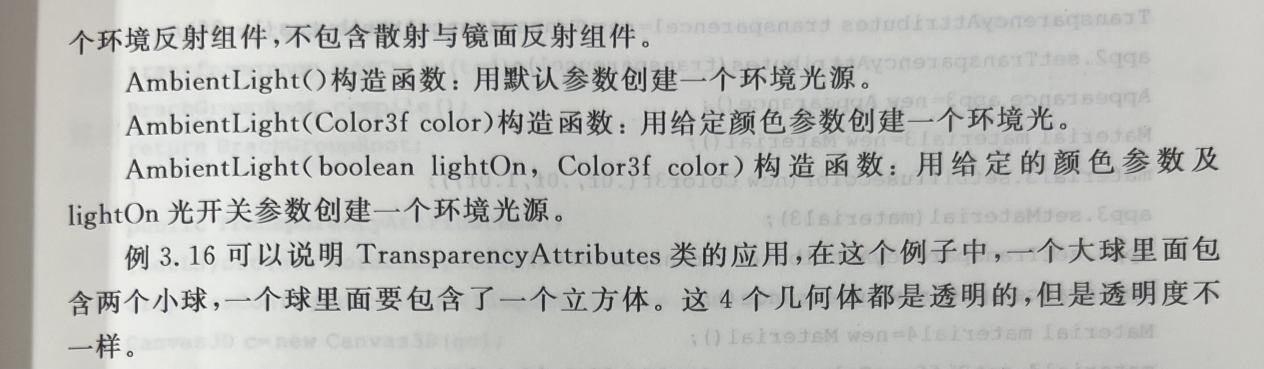
背景和光照：学会了如何设置背景颜色以及如何添加定向光源来影响模型的照明效果。这是创建逼真三维场景的关键。

鼠标交互：通过添加鼠标旋转、缩放和平移的交互功能，可以学习如何使用户能够以交互方式浏览和操作三维场景。

渲染和显示：最后，学会了如何将创建的四边面添加到场景中，并通过设置外观（Appearance）来定义模型的渲染属性。

**实验三总结：**





创建Java 3D应用程序：学习了如何创建一个Java 3D应用程序，包括设置窗口和创建3D场景。

透明物体和透明属性：了解了如何使用QuadArray类创建透明的三维物体，并设置不同的透明度，以实现透明效果。这是创建逼真的透明物体的关键。

环境光源：学会了如何添加环境光源，以改善整个场景的渲染效果。环境光源通常用于减轻阴影和强调物体的轮廓。

鼠标交互：通过添加鼠标旋转、缩放和平移的交互功能，可以学习如何使用户能够以交互方式浏览和操作三维场景。

渲染和显示：最后，学会了如何将创建的透明物体和环境光源添加到场景中，并通过设置外观（Appearance）来定义物体的渲染属性。